Eigenstructure of Maximum Likelihood from Counting Data

FANGHU DONG AND GUOLIANG TIAN

The University of Hong Kong

Abstract

The MLE (Maximum Likelihood Estimate) for a multinomial model is proportional to the data. We call such estimate an eigenestimate and the relationship of it to the data as the eigenstructure. When the multinomial model is generalized to deal with data arise from incomplete or censored categorical counts, we would naturally look for this eigenstructure between MLE and data. The paper finds the algebraic representation of the eigenstructure (put as Eqn (2.1), with which the intuition is visualized geometrically (Figures 2.2 and 4.3) and elaborated in a theory (Section 4). The eigenestimate constructed from the eigenstructure must be a stationary point of the likelihood, a result proved in Theorem 4.42. On the bridge between the algebraic definition of Eqn 2.1) and the Proof of Theorem 4.42, we have exploited an elementary inequality (Lemma 3.1) that governs the primitive cases, defined the *thick* objects of fragment and slice which can be assembled like mechanical parts (Definition 4.1), proved a few intermediary results that help build up the intuition (Section 4), conjectured the universal existence of an eigenestimate (Conjecture 4.32), established a criterion for boundary regularity (Criterion 4.37), and paved way (the Trivial Slicing Algorithm (TSA)) for the derivation of the *Weaver algorithms* (Section 5) that finds the eigenestimate by using it to reconstruct the observed counts through the eigenstructure; the reconstruction is iterative but derivative-free and matrix-inversion-free. As new addition to the current body of algorithmic methods, the Weaver algorithms craftily tighten threads that are weaved on a rectangular grid (Figure 2.3), and is one incarnation of the TSA. Finally, we put our method in the context of some existing methods (Section 6). Softwares are attached and demonstrated at http://hku.hk/jdong/eigenstruct2013a.html

1 Introduction

1.1. A metaphor. Suppose we have an array of chunks of certain material concealed in separate but identical boxes so that we cannot see their sizes nor weigh them on a scale. The material is a crystal, and, as usual, is made of atoms that are distributed homogeneously (with certain symmetry) inside the material. The boxes are labeled 1 to n. Although we don't have direct access to the chunks in the box, we do hear the screams of a tiny demon living freely inside the n chunks. The demon moves itself only by hopping from one place to another. It can hop from one chunk to another in a manner that is instant, impulsive, and free; it can also hop from one place to another inside the same chunk in the same manner that is instant, impulsive, and free. In short, to the demon, the n chunks' internal space is a connected piece dotted homogeneously with atoms labeled with 1 to n. To us, the only two important things to know are that the demon hops constantly and randomly in its living room—though sometimes a portion of the living room may be temporarily unavailable to the demon (due to our visitors' meddling with the boxes, for example)—and that, every time it collides itself at an atom of the crystal, the demon screams the label of the chunk out and we hear it—though we may not always hear the label clearly. Using the demon's screams, we would like to build a score for every chunk so that, through the scores, we gain a good sense of how the mass of each chunk compares to the masses of the others. To do this, we build a counting machine to count the demon's screams of each label. When the machine does not hear exactly which label the demon screams, it creates a grand-label that groups all labels possible to this scream and count it up by 1. When the machine sees that visitors are meddling with the boxes, it excludes those boxes being meddled with by creating a grand-label that groups all the unmeddled boxes and count negatively at this grand-label by the total number of demon screams during the meddling.

1.2. Using point sets to represent the chunks. Encapsulating this is the following class of phenomena: A generalized type of counts (areas, volumes, lengths, weights, scores, magnitudes, degrees, intensities, etc.) are recorded for an array of objects. We represent each of these objects as a point set, member of a set family, so that we can talk about the "unionic objects" and the "intersectional objects", and record their counts. The strong assumption that starts our modeling is that the points are indistinguishable, each equipped with a unit mass, and void of internal structure; and the total mass of the family is finite, so that we can pick the proper unit system to make this total equal to 1 and the masses are now probabilities. Each point set characterizes itself by its probability, gained from the points it is consisted of. Then it is the probability of each set we initially aim to deduce, using only the recorded counts of the labels screamed out by the demon as the only source of

information. In the maximum likelihood paradigm this means looking for the probability configuration with maximum likelihood.

1.3. Three types of counts and the question of their unifiability.

1.3.1. Ionic counts. If the recorded counts are the actual counts of events and let there be no two events occur in union, then this is the multinomial model (or Dirichlet model, depending on whether the bases or the exponents are fixed). In the multinomial/Dirichlet model the counts can be regarded as the *eigen* (direct, linear, proportional, true) manifestation of the underlying probability masses. This is because there exists a unique probability configuration that is both the maximum likelihood configuration and is on the line of the counts vector. We call the events in this first case the ionic events—these correspond to the labels exactly heard by us from the streaming demon.

1.3.2. Unionic counts. If the recorded counts include those of some unionic events (ionic events combined logically by OR), then we are compelled to answer the following question:

> *Can we reconstruct the actual counts in the similar way (the eigen way) from the maximum likelihood probability configuration?*

The unionic counts correspond to the labels heard ambiguously.

1.3.3. Conditional counts. The same question must be answered if we are in a third situation: some of the events are *excluded* during particular counting sessions, causing the probability to redistribute onto the conditional sample space. In this third case we call the events conditional events and these correspond to the labels heard when some boxes are being meddled with.

1.4. A very simple example introducing the new technique.
For example, we want to find the maximum likelihood probability configuration (p_1, p_2, p_3)[1]for the multinomial-like kernel:

$$x_1^2 x_2^2 x_3^2 \left(x_1 + x_2\right)^4 .$$

A simple manipulation will solve $(p_1, p_2, p_3) = (0.4, 0.4, 0.2)$.

The new technique to solve the MLE is eigenreconstruction. Briefly, it means generating the following system properly and solving it. The algebraic rules underpinning its generation is given in the next section (Eqn 2.1). For the moment, we prepare ourselves by asking if we can get a coinciding (p_1, p_2, p_3):

$$\begin{cases} x_1 a = 2 \\ x_2 a = 2 \\ x_3(a + b) = 2 \\ (x_1 + x_2)\, b = 4 \\ x_1 + x_2 + x_3 = 1 \end{cases}$$

For this example, the answer is yes (with $a = b = 5$). Eigenreconstructibility at least requires all the counts lie in the linear space spanned by the coordinates of the maximum likelihood probability vector. In a more geometric perspective, when we treat the coefficients as variables too, this is a system of polynomial equations, each having a total order equal to 2, and each of these polynomial equations defines a surface on which the points are the polynomial's roots. The intersection of these surfaces, called the algebraic variety defined by the polynomial system, is the solution set of the probability and the coefficients.

In general, the answer is probably still yes, and this makes the maximum likelihood probability configuration not only the weightiest, but also the intuitively convincing *because it has a structure*—this is the philosophy of this paper. The insight gained from the effort trying to understand eigenreconstructibility algebraically, computationally, and geometrically will lend explanation to questions like why the kernels $x_1^2 x_2^3 x_3^5 (x_1 + x_2)^{-4}$ and $x_1^2 x_2^3 x_3^{30}(x_1 + x_2)$ share the same maximum likelihood probability configuration $(\frac{1}{15}, \frac{1}{10}, \frac{5}{6})$.

---

[1]In this paper, we reserve the symbol p for the eigenestimate and, later, the intersection of the compatibility axis with the simplex, that is, p is the solution of the following system. We use x as the variable, presumably simplicial if not explicitly said, that is trying to achieve the eigenestimate p. The reason is, before we conclude anything, we have to always remind ourselves that the system need not be consistent and p need not exist.

2 Structure and Game

2.1. $\rho(E)$ and $p(E)$. Denote the *finite family of mutually exclusive sets* by $\mathcal{E} = \{E_i : i = 1, 2, \ldots, n\}$ and by $\mathcal{A} = \sigma\mathcal{E}$ the *collection of all possible unions* of sets in $\mathcal{E}$ (with the addition of $\emptyset$), which is, in this case, the σ-algebra generated by $\mathcal{E}$. Thus $(\bigcup \mathcal{E}, \sigma\mathcal{E})$ is a (finite) measurable space.

Denote the *generalized counts*[2] by $\rho : \mathcal{A} \to \mathbb{R}$, whence the recorded counts by $\rho(\bigcup \mathcal{F})$, for some subcollection $\mathcal{F} \subset \mathcal{E}$, abbreviated as ρ_i, ρ_{ij}, ρ_{ijk}, ..., like $\rho_1 = \rho(E_1)$, $\rho_{12} = \rho(E_1 \cup E_2)$, $\rho_{123} = \rho(E_1 \cup E_2 \cup E_3)$, ...

Denote the *probability* by $p : \mathcal{A} \to [0, 1]$ and with similar abbreviations we write $p_1 = p(E_1)$, $p_{12} = p(E_1 \cup E_2)$, $p_{123} = p(E_1 \cup E_2 \cup E_3)$, and so on.

Call E_is the **ionic events**, and union of more than one E_i's the **unionic events**. Thus the subscript shared by both ρ and p represents an event.

Remark 2.1. The connection to the familiar settings of categorical counts is the following. First, a category (ionic or unionic) is a container of (generalized) counts. An ionic category is a category having no non-empty subcategories other than itself. Here, a category is represented by a (measurable) set characterized by its probability. In the above E_i-$\mathcal{E}$-$\mathcal{A}$ notation, an ionic category is represented by an E_i, the array of all ionic categories are represented by $\mathcal{E}$, and the collection of all (ionic or unionic) categories is represented by $\mathcal{A}$. An element of $\mathcal{A}$ is E (subscript dropped) and $\rho(E)$ is the counts that fall into E. The extraordinary thing to note is that when $\rho(E)$ is negative, we say that E is counted negatively.

2.2. The existences of thick slices and the eigenstructure. ρ may take an additional underlined superscript ^{e} to announce its membership in a sub-product e called a *slice* possessing a property called *co-thickness*[3], a real number denoted by τ^e. For instance, $\rho_{12}^{(2)}$ denotes the part of the counts ρ_{12} that are allocated to the 2nd slice possessing co-thickness $\tau^{(2)}$.

With the earlier example, likelihood $x_1^2 x_2^2 x_3^2 (x_1 + x_2)^4$ consists of two slices $x_1^2 x_2^2 x_3$ and $(x_1 + x_2)^4 x_3$. We use the superscript notation to write $\rho_1^{(1)} = 2$, $\rho_2^{(1)} = 2$, $\rho_3^{(1)} = 1$, $\rho_3^{(2)} = 1$, $\rho_{12}^{(2)} = 4$ and $\tau^{(1)} = 5$, $\tau^{(2)} = 5$;[4] note that $0 = \rho_1^{(2)} = \rho_2^{(2)} = \rho_{12}^{(1)}$ as the three events 1, 2, and 12 each has membership in only one slice. An illustration is the diagram in Figure 2.1.
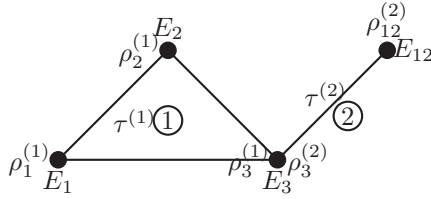

Figure 2.1: Illustrating the notation of ρ by a two-slice, four-event likelihood

The algebraic understanding of the relationship between ρ and p reduces to two rules:

$$\begin{cases} \rho_E^e \equiv \tau^e p_E \\ \rho_E \equiv \sum_e \rho_E^e \end{cases} \tag{2.1}$$

for all $E \in \mathcal{A}$. The symbol e still denotes a slice and each E pertains to at least an e. The summation in the second rule is carried over all slices that E is pertaining to. These rules represent the structural relationship between ρ and p. We name the relationship by calling ρ an **eigenreconstruction** of p and p the **eigenestimate** based on ρ.

These two seemingly simple algebraic rules surprise us by implying that, when the ρ's are fixed by observations, any satisfactory $p : \mathcal{A} \to \mathbb{R}$ is a stationary point of the likelihood given by the ρ's, an analytic property. (Theorem 4.42)

A geometric understanding starts from observing that τ^e is void of a subscript denoting events, which means it is one thickness across all components E in the slice e. An illustration using the same likelihood $x_1^2 x_2^2 x_3^2 (x_1 + x_2)^4$ as before is given in Figure 2.2. (A more extensive illustration is Figure 4.3)

---

[2]In the sense not only that the counts need not be integers, but also that the counts can be negative—in that case it is also a generalized measure under the physical image of an electric charge.

[3]Exact meaning of *slice* and its various properties including *co-thickness* will be developed in a later section.

[4]The 2 co-thicknesses are equal by coincidence. Computing co-thickness will be explained later.
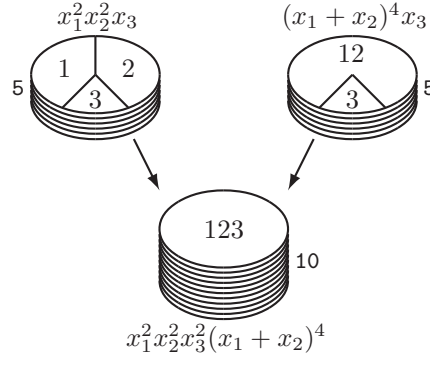
Figure 2.2: Geometric understanding of the relationship between ρ and p

2.3. Realistic likelihoods and a Game of Optimization. The following examples are demonstrated at http://hku.hk/jdong/eigenstruct2013a.html.

Example 2.2. *Five players are playing a series Ping-Pong matches. Their first 6 match scores are show in Table 2.1. Based on the match scores (ρ), it is required to determine a final score for each of the five players (normalizing the final scores gives the p).*

Table 2.1: n-ary comparison from paired scores from Ping-Pong matches

Match	A-B	C-D	A-E	B-C	D-E	A-D	......
Score	21-16	18-21	19-21	25-27	22-20	21-18	......

Based on the Bradley-Terry model, with $\sum x_i = 1$, the likelihood is

$$\frac{x_1^{21} x_2^{16}}{(x_1 + x_2)^{37}} \cdot \frac{x_3^{18} x_4^{21}}{(x_3 + x_4)^{39}} \cdot \frac{x_1^{19} x_5^{21}}{(x_1 + x_5)^{40}} \cdot \frac{x_2^{25} x_3^{27}}{(x_2 + x_3)^{52}}$$
$$\cdot \frac{x_4^{22} x_5^{20}}{(x_4 + x_5)^{42}} \cdot \frac{x_1^{21} x_4^{18}}{(x_1 + x_4)^{39}} \cdot \cdots .$$

Example 2.3. *A search engine provider wants to estimate the share of trust of global web users by 5 types of information websites: online encyclopedias, online newspapers, web forums, personal blogs, online marketing sites. They record data in the following way. Every time a search result page contains some of the 5 types: those shown types that are clicked receives a score 5,4,3,2,1, in the time order of clicks; those types that are shown but not clicked will score 0; those remaining types that are not shown will be excluded in the particular record. Table 2.2 lists the first 4 entries of data (that potentially grow unboundedly) gathered on the web.*

Table 2.2: Web source comparison

#	Ency.	News	Forum	Blog	Mkt.
1	5	4	-	-	0
2	-	5	4	3	-
3	-	3	-	4	5
4	5	-	0	0	0
...	...	...	...	...	...

The likelihood can be written as (with $\sum x_i = 1$)

$$\frac{x_1^5 x_2^4}{(x_1 + x_2 + x_5)^9} \cdot \frac{x_2^5 x_3^4 x_4^3}{(x_2 + x_3 + x_4)^{12}} \cdot \frac{x_3^3 x_4^4 x_5^5}{(x_2 + x_4 + x_5)^9} \cdot \frac{x_1^5}{(x_1 + x_3 + x_4 + x_5)^5} \cdot \cdots .$$

Example 2.4. *Frequencies (ρ) over the cross-classification by a pair of binary variables are sampled from a population. But the raw frequency data not only contain frequencies of complete (ionic) classifications, but also of incomplete (unionic) classifications and conditional classifications. It is required to use all of them to determine the relative frequency (normalizing to the p-measure) of each of the 4 categories. An example is given as the contingency tables in Table 2.4.*

The likelihood can be written as (with $\sum x_i = 1$)

$$x_1^{17} x_2^{29} x_3^{24} x_4^{15} \cdot (x_1 + x_2)^{12} (x_3 + x_4)^8 (x_1 + x_3)^{24} (x_2 + x_4)^{20}$$
$$\cdot \frac{x_2^8 x_3^{14}}{(x_2 + x_3)^{22}} \frac{x_2^4 x_4^2}{(x_2 + x_4)^6} \frac{x_1^5}{(x_1 + x_4)^5} \frac{x_1 x_3^3}{(x_1 + x_3 + x_4)^3}$$

4

Table 2.3: Three types of Categories

Ionic categories with Unionic categories on two margins

	A=0	A=1	A=x
B=0	17	29	12
B=1	24	15	8
B=x	24	20	

Conditional categories

	A=0	A=1
B=0	-	8
B=1	14	-

	A=0	A=1
B=0	-	4
B=1	-	2

	A=0	A=1
B=0	5	-
B=1	-	0

	A=0	A=1
B=0	1	-
B=1	2	0

simplifying to

$$\frac{x_1^{23} x_2^{41} x_3^{40} x_4^{17} \left(x_1 + x_2\right)^{12} \left(x_3 + x_4\right)^8 \left(x_1 + x_3\right)^{24} \left(x_2 + x_4\right)^{14}}{\left(x_2 + x_3\right)^{22} \left(x_1 + x_4\right)^5 \left(x_1 + x_3 + x_4\right)^3}.$$

For this likelihood, we give our first illustration of how the structural understanding can be applied to find the maximum likelihood probabilities. To optimize, we will play the game laid out as in Figure 2.3. The bottom and right margins place the counts while the center 0-1 weaving grid topped with a row of straight zeros specifies the matrix of the game which alludes to the pattern of cracking of the sample space. The top and left margins are the numbers to be filled consistently by the following 3 rules:

(**Rule** *1) Every "?" of the top margin when multiplied by the sum of those "?"s of the left margin filtered by a "0" in that column headed by the original "?" of the top margin must equal to the corresponding count at the bottom margin of that column.*

(**Rule** *2) Every, except the leading, "?" on the left margin when multiplied by the sum of those "?"s of the top margin filtered by a "1" in that row led by the the left "?" must equal to the corresponding count on the right margin of that row.*

(**Rule** *3) The top-margin "?"s must sum to 1.*

	?	?	?	?	
?	0	0	0	0	
?	1	1	0	0	12
?	0	0	1	1	8
?	1	0	1	0	24
?	0	1	0	1	14
?	0	1	1	0	-22
?	1	0	0	1	-5
?	1	0	1	1	-3
	23	41	40	17	

Figure 2.3: Play board of the optimization game

By the time one consistently fills all the "?"s, the top-margin "?"s will reveal the maximum-likelihood probabilities. (For your interest the answer to this game is Figure 2.4.) Here is how we solve it. First we initialize the top row to a good guess: proportional to the bottom-margin counts while summing to 1. (1) Then we are able to determine all the left-margin "?"s except the left-top "?" by Rule 2. (2) Then we determine the left-top "?" by the formula of Lemma 5.1 below.[5] (3) Then we determine the top-margin "?"s by Rule 1 (with normalization to comply with Rule 3). These 3 steps complete an iteration. With just 10 iterations we will solve the problem with a good precision: 10^{-9} by sum of squared deviation used in Lemma 5.1. We see this will not be

---

[5] In Lemma 5.1's notation, $-\frac{b(x)}{2a(x)}$ is what we will here use to fill the left-top "?", which is a value of the variable denoted τ_0 there—so it is in fact the number that minimizes a global error measure that is in quadratic relationship with τ_0. More notations: x there is the top-margin "?"s here; $\tau(x)$ there is the left-margin "?"s except the left-top "?" here; Δ there is the weaving matrix (removing the top-row zeros), transposed, here; $\mathbf{1}_{n \times q}$ there is an all-one matrix; a (not $a(x)$) there is the bottom margin here.

a computationally expensive procedure at all! In fact, the convergence is linear, i.e., every iteration reduces the error by an order, and that's how just 10 iterations can we get from a starting error of 10^1 down to 10^{-9}, and if we continue, 10 more iterations will bring precision to 10^{-19}.

3   Inequality

Lemma 3.1. *For positive real numbers $x_1, \ldots, x_n > 0$ and $a_1, \ldots, a_n > 0$ we have*

$$\prod_{i=1}^{n} x_i^{a_i} \leqslant \frac{\prod_{i=1}^{n} a_i^{a_i}}{\left(\sum_{i=1}^{n} a_i\right)^{\sum_{i=1}^{n} a_i}} \left(\sum_{i=1}^{n} x_i\right)^{\sum_{i=1}^{n} a_i} \tag{3.1}$$

attaining equality iff (if and only if) there exists a common positive ratio $k > 0$ such that for each i it holds $\frac{x_i}{a_i} = k$.

Proof. (Work with $\frac{x_i}{a_i}$ and connect to the *Weighted AM-GM Inequality*, with its equality condition). Rewrite the target inequality as

$$\prod_{i=1}^{n} a_i^{a_i} \prod_{i=1}^{n} \left(\frac{x_i}{a_i}\right)^{a_i} \leqslant \frac{\prod_{i=1}^{n} a_i^{a_i}}{\left(\sum_{i=1}^{n} a_i\right)^{\sum_{i=1}^{n} a_i}} a_i^{\sum_{i=1}^{n} a_i} \left(\sum_{i=1}^{n} \frac{x_i}{a_i}\right)^{\sum_{i=1}^{n} a_i}$$

$$\Leftrightarrow \prod_{i=1}^{n} \left(\frac{x_i}{a_i}\right)^{a_i} \leqslant \frac{a_i^{\sum_{i=1}^{n} a_i}}{\left(\sum_{i=1}^{n} a_i\right)^{\sum_{i=1}^{n} a_i}} \left(\sum_{i=1}^{n} \frac{x_i}{a_i}\right)^{\sum_{i=1}^{n} a_i} = \left(\sum_{i=1}^{n} \frac{a_i}{\sum_{i=1}^{n} a_i} \frac{x_i}{a_i}\right)^{\sum_{i=1}^{n} a_i}$$

$$\Leftrightarrow \prod_{i=1}^{n} y_i^{\frac{a_i}{\sum_{i=1}^{n} a_i}} \leqslant \sum_{i=1}^{n} \frac{a_i}{\sum_{i=1}^{n} a_i} y_i \tag{3.2}$$

$$\Leftrightarrow \prod_{i=1}^{n} y_i^{w_i} \leqslant \sum_{l=1}^{n} w_i y_i \tag{3.3}$$

where we used substitution $y_i = \frac{x_i}{a_i}$ in Eqn (3.2) and another substitution $w_i = \frac{a_i}{\sum_{i=1}^{n} a_i}$ in Eqn (3.3). The latter is the *Weighted AM-GM Inequality*.

It is crucial that we now check and confirm that all equalities can hold jointly iff $\frac{x_i}{a_i} = k$ for all i, given the existence of such a uniform constant k which must be positive.

For the Weighted AM-GM Inequality, there are two classical proofs: The first one due to G. Pólya links it with the fundamental inequality $e^x \geq 1 + x$ for any positive number; the second one takes a logarithm of the inequality and uses the fact that the logarithm function satisfies Jensen's Inequality. □

Remark 3.2. Lemma 3.1 should be understood in two aspects. First is an understanding towards the equality attainment condition: Given the power vector a, and the functional form of the left hand side $\prod_{i=1}^{n} x_i^{a_i}$ of which

Figure 2.4: Answer to Figure 2.3

	0.2288	0.3126	0.3153	0.1433	
87.41	0	0	0	0	
22.17	1	1	0	0	12
17.44	0	0	1	1	8
44.11	1	0	1	0	24
30.71	0	1	0	1	14
-35.04	0	1	1	0	-22
-13.44	1	0	0	1	-5
-4.36	1	0	1	1	-3
	23	41	40	17	

the base vector x is variable, then if we choose to put x on the same line as a, then we have all the inequalities become equalities, to equate the left hand side's product form to the right hand side's summation form, and vice versa. Second is to notice that the fraction prefixing the right hand side is a function solely of the a's; hence the inequality describes the increase of value when some ions in the products form union, or equivalently, the decrease of value when some unions are split into ions.

Example 3.3. *We demonstrate the intuition behind Lemma 3.1's formalism by showing two integer power examples.*

a. $(x_1 + x_2)^5 \geqslant \frac{5^5}{3^3 2^2} x_1^3 x_2^2$. *This is because*

$$x_1^3 x_2^2 = \frac{x_1}{3}\frac{x_1}{3}\frac{x_1}{3}\frac{x_2}{2}\frac{x_2}{2} 3^3 2^2 \leqslant 3^3 2^2 \left(\frac{3\frac{x_1}{3} + 2\frac{x_2}{2}}{3+2}\right)^{3+2} = 3^3 2^2 \left(\frac{x_1 + x_2}{5}\right)^5$$

The equality is attained iff (x_1, x_2) is co-linear with $(3, 2)$.

b. $(x_1 + x_2)^7 x_3^3 x_4^5 \leqslant \frac{3^3 5^5 7^7}{15^{15}} (x_1 + x_2 + x_3 + x_4)^{15}$. *This is because*

$$(x_1 + x_2)^7 x_3^3 x_4^5 \leqslant 3^3 5^5 7^7 \left(\frac{3\frac{x_1}{3} + 5\frac{x_2}{5} + 7\frac{x_3+x_4}{7}}{3+5+7}\right)^{3+5+7}$$

The equality is attained iff $(x_1 + x_2, x_3, x_4)$ is co-linear with $(7, 3, 5)$. More importantly, together with the inequality in the previous example, the two equalities are jointly attained iff (x_1, x_2, x_3, x_4) is co-linear with $(21, 14, 15, 25)$.

Corollary 3.4. *If we require $\sum\limits_{i=1}^{n} x_i = \sum\limits_{i=1}^{n} a_i = 1$ in Lemma 3.1 then*

$$\prod_{i=1}^{n} x_i^{a_i} \quad \leqslant \quad \prod_{i=1}^{n} a_i^{a_i} \tag{3.4}$$

$$\sum_{i=1}^{n} a_i \ln x_i \quad \leqslant \quad \sum_{i=1}^{n} a_i \ln a_i \tag{3.5}$$

and the equalities are attained iff $x = a$.

Remark 3.5. The logarithmic version Eqn (3.5) is the (finite) entropy inequality.

Corollary 3.6. *Let $x \in (0, +\infty)^n$ be a vector of length n positive reals. Let $\delta \in \{0,1\}^n$ be a vector of n bits. Let $\beta \in [0, +\infty)^n$ be a non-zero vector of n non-negative reals such that $\beta_j = 0$ if $\delta_j = 0$. Let $b = \sum\limits_{i=1}^{n} \beta_i > 0$. Let $0^0 := 1$. Then*

$$(\delta^\intercal x)^b \geqslant \frac{b^b}{\prod\limits_{i=1}^{n} \beta_i^{\beta_i}} \prod_{i=1}^{n} x_i^{\beta_i}$$

attaining equality iff there exists a positive k such that $\frac{x_i}{\beta_i} = k$ for each of those i's having $\delta_i = 1$.

Example 3.7. *Let $n = 5$, $\delta = (1,0,1,0,1)^\intercal$, $\beta = (3,0,4,0,6)^\intercal$, $b = 3+0+4+0+6 = 13$. Then $\forall x \in (0, +\infty)^n$, we have*

$$(1x_1 + 0x_2 + 1x_3 + 0x_4 + 1x_5)^{13} \geqslant \frac{13^{13}}{3^3 0^0 4^4 0^0 6^6} x_1^3 x_2^0 x_3^4 x_4^0 x_5^6$$

attaining equality iff $x_1 : x_3 : x_5 = 3 : 4 : 6$.

4 Theory

The theory is for the understanding of the conditions surrounding maximization. It analytically realizes the structural relationship between ρ and p. The theory culminates in a theorem proving that any p satisfying the two algebraic rules of Eqn (2.1) is a stationary point of the likelihood. The theory also leaves two conjectures for future work. Before we start the development, we lay out the map of the theory as in Figure 4.1.

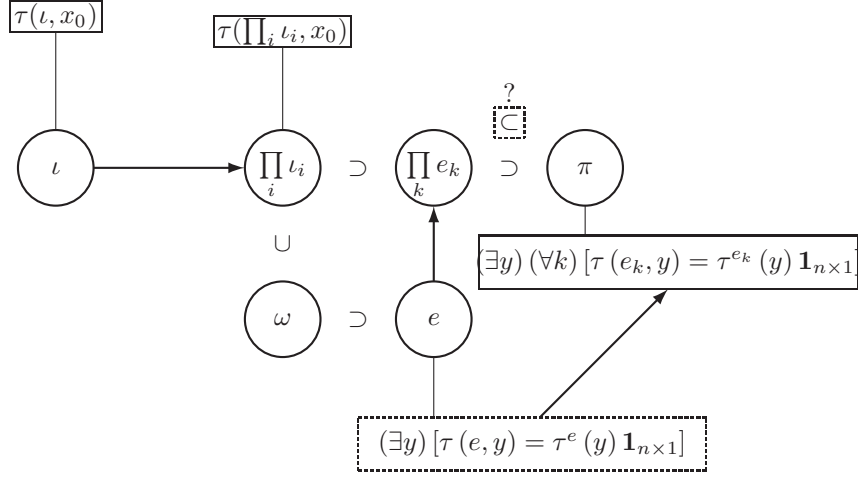**4.1. Fragment, Slice, and Superposition.**

Figure 4.1: Map of Theory

4.1.1. Three Objects and some Basic Properties.

Definition 4.1 (Fragment, Slice, and Superposition). Let n and q be two positive integers. Let $x \in (\mathbb{R}\backslash\{0\})^n$ be a vector of length n whose coordinates are non-zero real numbers. Let $\Delta = [\delta_1, \ldots, \delta_q] \in (\{0,1\}^n \backslash \{0\}^n)^q$ be a sans-zero-column matrix of q columns each of which is a vector of n bits, representing an (ion or union) event . Let $\rho \in (\mathbb{R}\backslash\{0\})^q$ be a vector of length q whose coordinates are non-zero real numbers. Note that ρ_j, the j-th component of ρ, is the count of the event represented by δ_j. Denote by $\mathbf{1}_{s \times t}$ the $s \times t$ matrix of 1s.

a. (*order-1 fragment*) The power expression
$$\iota = (\delta_1^\mathsf{T} x)^{\rho_1}$$
is an *order*-1 *x-fragment*. We call the non-zero vector δ_1 the *event pattern* of ι, the positive integer $\delta_1^\mathsf{T} \mathbf{1}_{n \times 1}$ the *event size* of ι, and the exponent ρ_1 the *event count* of ι. An order-1 x-fragment is an *ionic x-fragment* iff its event pattern is a standard unit vector.

b. (*union of order-1 fragments*) Let $\iota_1 = (\delta_1^\mathsf{T} x)^{\rho_1}$ and $\iota_2 = (\delta_2^\mathsf{T} x)^{\rho_2}$ be two order-1 x-fragments, $\iota_1 \cup \iota_2 = (\max(\delta_1, \delta_2)^\mathsf{T} x)^{\rho_1 + \rho_2}$ is called their *union*, where the max operation is component-wise. When $\max(\delta_1, \delta_2) = \mathbf{1}_{n \times 1}$, we call it an *exhaustive* union of order-1 fragments.

c. (*order-q fragment*) A product of q order-1 fragments
$$\omega = \prod_{j=1}^{q} (\delta_j^\mathsf{T} x)^{\rho_j}$$
is an *order-q x-fragment* iff it satisfies a **closure condition** expressed in the following two equivalent versions:

   1. (closure condition version 1) $\forall i \neq k, \delta_i + \delta_k \in \{0,1\}^n$
   2. (closure condition version 2) $\Delta \mathbf{1}_{n \times 1} \in \{0,1\}^n$

We call the real vector x *the vector of ions* of ω. We call the number q the *order* of ω. We call the vector ρ the *event counts* of ω. We call the $n \times q$ bit matrix $\Delta = [\delta_1, \ldots, \delta_q]$ the *event pattern* of ω. We often write $\omega(x)$ or $\omega(x|\rho, \Delta)$ to be explicit about its nature as a function.

d. (*slice*) An x-fragment
$$e = \prod_{j=1}^{q} (\delta_j^\mathsf{T} x)^{\rho_j}$$
is an *x-slice* iff the following **exhaustiveness** is satisfied, i.e., $\sum_{j=1}^{q} \delta_j = \mathbf{1}_{n \times 1}$ or equivalently $\Delta \mathbf{1}_{n \times 1} = \mathbf{1}_{n \times 1}$.

We often write $e(x)$ or $e(x|\rho, \Delta)$ to be explicit about its nature as a function. For a point $p \in (\mathbb{R}\backslash\{0\})^n$, the slice e is said to *achieve co-thickness* at p iff $\forall j$, $\exists \tau^e \neq 0$, such that $\rho_j = \tau^e \delta_j^\mathsf{T} p$. Such a $\tau^e = \tau^e(p)$ is called the *co-thickness* of e at p.

8

e. (*superposition*) A product of slices

$$\pi = \prod_k e_k$$

is a superposition iff $\exists p \in (\mathbb{R} \backslash \{0\})^n$ such that every slice e in the product achieves co-thickness at cp, $\forall c \neq 0$. We often write $\pi(x)$ or $\pi(x|\rho, \Delta)$ to be explicit about its nature as a function. The line represented by cp is called a *compatibility axis* of π. Any slice is a product of only one slice, hence may be a superposition on its own.

Remark 4.2. We often deal with the case that x is a point on the simplex. For that case we may add to the name a modifier "simplicial", whose logical attachment is solely with x, though we allow lexical associativity so that the term "(simplicial x)-fragment" means the same as the term "simplicial (x-fragment)". The restricted version of the notion inherits all properties from the unrestricted version, and may have its own properties. It is important to understand that the restriction by a simplex is really to intersect the compatibility axis by the simplex to result in a single point. Thus, even the intersecting geometric object is not a simplex, the result developed here abiding to the whole compatibility axis is still inheritable. There is no need to carry along the restriction from the beginning until the solution, but to only apply the restriction after the whole compatibility axis is solved.

Remark 4.3. A nuisance matter here is the inclusion/exclusion of zero at certain coordinates of x and ρ; we choose to exclude them all. We think of a slice as a joint probability on a coarsened version of the original finite space of complete categorization. We think of the max operation on a few δ_js as the union of those sets as categories that each bit vector δ_j serves as indicator for inclusion. On the relation between an x-fragment and an x-slice, we would say "an x-fragment is *in* an x-slice" and "an x-fragment *of* an x-slice" if the former factorizes the latter.

Remark 4.4. Note that the order property is not a function. Any order-1 fragment is also an order-q fragment. The definitions of order-1 and order-q fragments does not define two disjoint sets; in fact, one of them is a subset of the other. For example, the order-1 fragment $(\delta_1^\intercal x)^5$ when written as $(\delta_1^\intercal x)^2 (\delta_1^\intercal x)^2 (\delta_1^\intercal x)$ is an order-3 fragment. The order property does become single-valued when we address the collected form of any fragment. The collection is over the same event patterns. Therefore we may also conceptualize the notion of "collected order".

Example 4.5. *Let $n = 5$ and $x = (x_1, x_2, x_3, x_4, x_5)^\intercal$:*

a. $\omega_1 = x_1$ *is an order-1 fragment (collected); further, it is an ionic fragment.*

- $q = 1$, $\delta_1 = (1, 0, 0, 0, 0)^\intercal$, $\rho_1 = 1$

b. $\omega_2 = x_1 x_2^2$ *is an order-2 fragment and it can not be an order-1 fragment.*

- $q = 2$, $\delta_1 = (1, 0, 0, 0, 0)^\intercal$, $\rho_1 = 1$, $\delta_2 = (0, 1, 0, 0, 0)^\intercal$, $\rho_2 = 2$
- *closure:* $\delta_1 + \delta_2 = (1, 1, 0, 0, 0)^\intercal$

c. $\omega_3 = x_1 x_2^2 (x_3 + x_4 + x_5)^3$ *is an order-3 fragment; further it is a slice*

- $q = 3$, $\delta_1 = (1, 0, 0, 0, 0)^\intercal$, $\rho_1 = 1$, $\delta_2 = (0, 1, 0, 0, 0)^\intercal$, $\rho_2 = 2$, $\delta_3 = (0, 0, 1, 1, 1)^\intercal$, $\rho_3 = 3$.
- *closure and exhaustiveness:* $\delta_1 + \delta_2 + \delta_3 = (1, 1, 1, 1, 1)^\intercal$

d. $\omega_4 = \frac{1}{(x_3 + x_4 + x_5)^2}$ *is an order-1 fragment but it is not ionic because its event pattern is not a standard unit vector.*

- $q = 1$, $\delta_1 = (0, 0, 1, 1, 1)^\intercal$, $\rho_1 = -2$

e. $\omega_5 = \frac{x_2^2 x_4^5}{(x_1 + x_3 + x_5)^3}$ *is an order-3 fragment; further, it is a slice*

- $q = 3$, $\delta_1 = (0, 1, 0, 0, 0)^\intercal$, $\rho_1 = 2$, $\delta_2 = (0, 0, 0, 1, 0)^\intercal$, $\rho_2 = 5$, $\delta_3 = (1, 0, 1, 0, 1)^\intercal$, $\rho_3 = -3$.
- *closure and exhaustiveness:* $\delta_1 + \delta_2 + \delta_3 = (1, 1, 1, 1, 1)^\intercal$

f. $\omega_6 = \frac{1}{x_1 + x_3 + x_4 + x_5}$ *is an order-1 fragment and it is $\omega_1 \cup \omega_4$*

- $q = 1$, $\delta_1 = (1, 0, 1, 1, 1)^\intercal = max[(1, 0, 0, 0, 0)^\intercal, (0, 0, 1, 1, 1)^\intercal]$, $\rho_1 = -1 = 1 + (-2)$

The followings are not x-fragments because each of them violates closure:

a. $\pi_1 = \frac{x_1^4 x_2^6 x_3^8 x_4^{10} x_5^{12} (x_1+x_2)^5 (x_3+x_4+x_5)^{15}}{(x_1+x_2+x_3)^9 (x_4+x_5)^{11}}$ *is not a slice but it is a product of 3 slices. It is also a superposition with compatibility axis* $c[2,3,4,5,6]^{\mathsf{T}}$.

- $q = 9$, $\Delta = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$,

- $\rho = \begin{bmatrix} 4 & 6 & 8 & 10 & 12 & 5 & 15 & -9 & -11 \end{bmatrix}^{\mathsf{T}}$

- *but* $\Delta \mathbf{1}_{9 \times 1} = \begin{bmatrix} 3 & 3 & 3 & 3 & 3 \end{bmatrix}^{\mathsf{T}}$

- $e_1 = x_1^4 x_2^6 x_3^8 x_4^{10} x_5^{12}$, $e_2 = (x_1 + x_2)^5 (x_3 + x_4 + x_5)^{15}$, $e_3 = (x_1 + x_2 + x_3)^{-9} (x_4 + x_5)^{-11}$.

b. $x_1 x_2^2 (x_1 + x_2)$

- $q = 3$, $\delta_1 = (1,0,0,0,0)^{\mathsf{T}}$, $\rho_1 = 1$, $\delta_2 = (0,1,0,0,0)^{\mathsf{T}}$, $\rho_2 = 2$, $\delta_3 = (1,1,0,0,0)^{\mathsf{T}}$, $\rho_3 = 1$

- *but* $\delta_1 + \delta_2 + \delta_3 = (2,2,0,0,0)^{\mathsf{T}}$

4.1.2. Two Partial Orderings.

Definition 4.6 (Covering). Inheriting notations from Definition 4.1:

a. (*order-1 covering*) Let $\iota_1 = (\delta_1^{\mathsf{T}} x)^{\rho_1}$ be an order-1 x-fragment. An order-1 x-fragment $\iota_2 = (\delta_2^{\mathsf{T}} x)^{\rho_2}$ is a *covering fragment* of ι_1 iff $\delta_1 \geqslant \delta_2$ (component-wise). We also say ι_2 *covers* ι_1.

b. (*order-q covering*) Let ω be an x-fragment of order $q \geqslant 2$. An x-fragment ξ of the same order q is a covering fragment of ω iff every order-1 x-fragment of ω has a **distinct** covering order-1 fragment among ξ's q order-1 x-fragments. We also say ξ *covers* ω.

c. (*collectability*) Two x-fragments ω and ξ are collectable fragments to each other iff they share the same set of event patterns. We also say that ω *collects with* ξ.

Remark 4.7. The event counts ρ is irrelevant to the definition of a covering fragment. Two fragments of different orders do not cover each other. Distinct correspondence between all order-1 fragments of ω and all order-1 fragments of ξ means there exists a bijection between the two sets. We will need this concept when extricating from divergence wrecked by the zero coordinates and negative exponentiation. We allow the following statement to hold: any fragment covers itself.

Lemma 4.8. *Inheriting notations from Definition 4.1:*

a. *"collects with" is an equivalence relation on the set of all x-fragments.*

b. *Covering is a partial ordering on the set of all x-fragments with respect to the equivalence relation "is collectable with" on the same set.*

c. *Two x-fragments are collectible with each other iff they cover each other.*

Proof. The first statement is trivial because the usual equality between two matrices underpins the notion. The third statement is implied in the second's anti-symmetry requirement. We show the second statement as following. A partial ordering requires reflexivity, anti-symmetry, and transitivity. Let $\omega_1 = (\delta_1^{\mathsf{T}} x)^{\rho_1}$, $\omega_2 = (\delta_2^{\mathsf{T}} x)^{\rho_2}$, $\omega_3 = (\delta_3^{\mathsf{T}} x)^{\rho_3}$ be any three order-1 x-fragments. Reflexivity is trivial: say for ω_1, we have $\delta_1 \geqslant \delta_1$ (component-wise). Anti-symmetry refers to (ω_1 covers ω_2 and ω_2 covers ω_1) $\implies$ ω_1 collects with ω_2, which is true because the condition means ω_1 and ω_2 share the same event pattern. Transitivity is trivial: ($\delta_1 \geqslant \delta_2$ and $\delta_2 \geqslant \delta_3$) $\implies$ $\delta_1 \geqslant \delta_3$. $\square$

Notation 4.9. We shall write $\omega_1 \overset{cover}{\geqslant} \omega_2$ or $\omega_2 \overset{cover}{\leqslant} \omega_1$ iff ω_1 covers ω_2. We shall write $\omega_1 \overset{cover}{=} \omega_2$ iff ω_1 collects with ω_2.

Example 4.10.

a. $(x_1 + x_2)^3 \overset{cover}{=} (x_1 + x_2)^{-5}$, $x_1^{30} \overset{cover}{\leqslant} (x_1 + x_2)^3$, $x_1^4 x_2^5 (x_1 + x_3) \overset{cover}{=} \frac{x_1 x_2}{x_1 + x_3}$ $\overset{cover}{\geqslant} \frac{1}{x_1 x_2 x_3}$.

b. $(x_1 + x_2 + x_3 + x_4 + x_5)^{-70} \overset{cover}{\geqslant} (x_1 + x_2 + x_3 + x_4)^7 \overset{cover}{\geqslant} (x_1 + x_2 + x_3)^{-70} \overset{cover}{\geqslant} (x_1 + x_2)^7$ *but none of them covers* $x_1^{-70} x_2^{-70}$ *because* x_1^{-70} *and* x_2^{-70} *do not find distinct covering order-1 fragments in any of them.*

Definition 4.11 (Refinement)**.** Let $\Omega = \prod\limits_{j=1}^{q} (\delta_j^\intercal x)^{\rho_j}$ be a product of q order-1 x-fragments. Another product of r order-1 x-fragments $\Xi = \prod\limits_{i=1}^{r} (\gamma_i^\intercal x)^{\sigma_i}$ is said to *refine* Ω iff $\forall j \in \{1, \ldots, q\}, \exists I_j \subset \{1, \ldots, r\}$, such that all the following four conditions are satisfied:

a. $I_j \neq \emptyset$

b. $j_1 \neq j_2 \implies I_{j_1} \bigcap I_{j_2} = \emptyset$

c. $\sum\limits_{i \in I_j} \gamma_i = \delta_j$

d. $\sum\limits_{i \in I_j} \sigma_i \geqslant \rho_j$

We also say that "Ξ *refines* Ω" and "Ω *coarsens* Ξ". When the last condition attain equalities for all j, then we say that "Ξ ***splits*** Ω."

Remark 4.12. While covering is a partial ordering on the set of all fragments, refinement is a partial ordering defined on the set of all products of order-1 fragments, and fragments, slices, and superpositions are all products of order-1 fragments. Refinement requires not only refining every event pattern but also dominating the event counts. Each of them can be a partial ordering but we do not bother to explicitly define them. We allow the following statement to hold: any fragment refines itself. The source of this concept traces from two places: First, we need it when we are dealing a product of slices involving negative counts while seeking a condition for the attainment of extremity along the compatibility axis; Second, the concept corresponds to features of the real sampling process when a category is split into a number of subcategories or its counts is increased.

Lemma 4.13. *Inheriting the notations from Definition 4.11:*

a. *Refinement is a partial ordering on the set of all products of order-1 x-fragments with respect to the usual equality between two functions as the equivalence relation.*

b. *Two products of order-1 x-fragments refines each other iff they are the same.*

c. *Splitting preserves total counts.*

Proof. The second statement is a consequence of the first's anti-symmetry requirement. We show the first statement. Reflexivity and Anti-symmetry are trivial because when comparing a fragment to itself we have: $q = r$, $I_j = \{j\}$, $\sum\limits_{i \in I_j} \gamma_i = \gamma_j = \delta_j$, and $\sum\limits_{i \in I_j} \sigma_i = \sigma_j = \rho_j$. Transitivity is neither hard to show: we just have one more nesting in the last 2 requirements. Let $\Omega = \prod\limits_{j=1}^{q} (\delta_j^\intercal x)^{\rho_j}$ refine $\Xi = \prod\limits_{i=1}^{r} (\gamma_i^\intercal x)^{\sigma_i}$ refine $\Psi = \prod\limits_{l=1}^{t} (\eta_l^\intercal x)^{\theta_l}$. For the former pair we continue to use the same "$\forall j \in \{1, \ldots, q\}, \exists I_j \subset \{1, \ldots r\}$" notion, and for the latter pair we use the analogous "$\forall l \in \{1, \ldots, r\}, \exists J_l \subset \{1, \ldots t\}$" notation so that $\sum\limits_{l \in J_i} \eta_l = \gamma_i$, hence $\sum\limits_{i \in I_j} \sum\limits_{l \in J_i} \eta_l = \delta_j$ which means each δ_j is the sum of a unique subset of the η_ls. This satisfies the third condition. For the last condition, from Ξ refining Ψ we have $\sum\limits_{l \in J_i} \theta_l \geqslant \sigma_i$ and from Ω refining Ξ we have $\sum\limits_{i \in I_j} \sigma_i \geqslant \rho_j$; eliminating ρ_j we have $\sum\limits_{i \in I_j} \sum\limits_{l \in J_i} \theta_l \geqslant \rho_j$. So the last condition hold and the proof is complete.

For Nr.3: Suppose Ξ splits Ω, then the total counts of Ω is $\sum\limits_{j=1}^{q} \rho_j = \sum\limits_{j=1}^{q} \sum\limits_{i \in I_j} \sigma_i$ which is just the total counts of Ξ. $\qquad\square$

Notation 4.14. We shall write $\Omega_1 \overset{refine}{\geqslant} \Omega_2$ or $\Omega_2 \overset{refine}{\leqslant} \Omega_1$ iff Ω_1 refines Ω_2. We simply write $\Omega_1 = \Omega_2$ iff Ω_1 and Ω_2 refines each other.

Example 4.15. *Referring to the earlier example of fragments:*

a. $\Xi_1 = x_1^2 x_2^4 (x_3 + x_4)^{-7} x_5^{100} \overset{refine}{\geqslant} \Omega_1 = x_1 x_2^2 (x_3 + x_4 + x_5)^3$

  • $q = 3$, $r = 4$, $\{I_1, I_2, I_3\} = \{\{1\}, \{2\}, \{3, 4\}\}$,

- $\sum\limits_{i \in I_1} \gamma_i = (1,0,0,0,0)^\intercal = \delta_1$, $\sum\limits_{i \in I_1} \sigma_i = 2 \geqslant 1 = \rho_1$,

- $\sum\limits_{i \in I_2} \gamma_i = (0,1,0,0,0)^\intercal = \delta_2$, $\sum\limits_{i \in I_2} \sigma_i = 4 \geqslant 2 = \rho_2$,

- $\sum\limits_{i \in I_3} \gamma_i = \gamma_3 + \gamma_4 = (0,0,1,1,1)^\intercal$, $\sum\limits_{i \in I_3} \sigma_i = \sigma_3 + \sigma_4 = -7 + 100 \geqslant 3 = \rho_3$

b. $\dfrac{x_1^3 x_2^4 x_3^5 x_1 x_2}{x_1 x_3 x_2 x_3} \overset{refine}{\geqslant} \dfrac{x_1^3 x_2^4 x_3^5 (x_1+x_2)^2}{(x_1+x_3)^4 (x_2+x_3)^5} \overset{refine}{\geqslant} \dfrac{(x_1+x_2+x_3)^4}{(x_1+x_3)^4 x_2 (x_2+x_3)^5 x_1}$
   $\overset{refine}{\geqslant} (x_1 + x_2 + x_3)^3$

c. $\xi_5 = (x_1 + x_3)(x_2 + x_4)^{100} x_5$ *does not refine* $\omega_5 = \dfrac{x_2^2 x_4^5}{(x_1 + x_3 + x_5)^3}$ *because although* $(x_1 + x_3) x_5 \overset{refine}{\geqslant} (x_1 + x_3 + x_5)^{-3}$, $(x_2 + x_4)^{100} \overset{refine}{\leqslant} x_2^2 x_4^5$.

4.2. More on Thickness. We have already encountered the notion of compatibility in the definition of superposition. We have also illustrated both the algebraic aspect and the geometric aspect of co-thickness. In a nutshell, compatibility is existence of a co-thickness; in a nutshell, co-thickness is uniform thickness of data across all events in the slice.

Notation 4.16. Denote by $\tau(\Omega, x_0)$ the x_0-thickness of Ω, a product of some order-1 fragments. $\tau(\Omega, x_0)$ is a vector, except when Ω is an order-1 fragment, in which case τ is a scalar.

Definition 4.17. Let $\Omega = \prod\limits_{j=1}^{q} (\delta_j^\intercal x)^{\rho_j}$ be a product of q order-1 x-fragment. The x_0-thickness of Ω is defined as

$$\tau(\Omega, x_0) := \left[\frac{\rho_1}{\delta_1^\intercal x_0}, \dots, \frac{\rho_q}{\delta_q^\intercal x_0} \right]^\intercal,$$

a column vector of q scalar components. When $q = 1$, $\tau(\Omega, x_0) = \frac{\rho_1}{\delta_1^\intercal x_0}$ is scalar-valued.

Lemma 4.18 (Inverse proportionality of thickness)**.** $\tau(\Omega, cx_0) = \frac{1}{c}\tau(\Omega, x_0)$, $\forall c \neq 0$.

Definition 4.19 (Compatibility between order-1 fragments)**.** Two order-1 fragments ι_1 and ι_2 are y-compatible iff there exists a point $y \in (\mathbb{R} \backslash \{0\})^n$ such that $\tau(\iota_1, y) = \tau(\iota_2, y)$.

Note 4.20. Due to (inverse) proportionality, we see that if ι_1 and ι_2 are y-compatible, then they are cy-compatible, $\forall c \neq 0$.

Definition 4.21 (Co-thickness of a slice and Superposability of slices)**.** Let $e = \prod\limits_{j=1}^{q} (\delta_j^\intercal x)^{\rho_j}$ be an x-slice. If there exists a point $y \in (\mathbb{R} \backslash \{0\})^n$ such that all order-1 fragments in e has the same y-thickness, then this y-thickness is a *co-thickness* of e, denoted by $\tau^e(y)$. That is, $\frac{\rho_1}{\delta_1^\intercal y} = \cdots = \frac{\rho_q}{\delta_q^\intercal y} = \tau^e(y)$. The slice e is said to *achieve co-thickness* at y. Two slices are *y-superposable* iff there exists a point $y \in (\mathbb{R} \backslash \{0\})^n$ at which both slices achieve co-thickness.

Notation 4.22. Denote by $e_1 \overset{y}{\diamond} e_2$ that the slices e_1 and e_2 are y-superposable.

Lemma 4.23. $\overset{y}{\diamond}$ *is an equivalence relation over the set of all x-slices, for some given $y \in \{\mathbb{R} \backslash \{0\}\}^n$.*

Remark 4.24. A co-thickness is a scalar-valued property for a slice. The prefix 'co-' stands for 'compatible'. In prefixing 'thickness', it basically equals 'iso-'. Regarding the correspondence between co-thickness and maximization, it seems that nature has preferences for the equitable, homogeneous allocation of count thickness over the events.

Lemma 4.25. *A product of order-1 fragments is a superposition **only if**, after collecting every class of collectible order-1 fragments, the matrix taking all the δ's as the columns has its every row summing to at least 1.*

Proof. Let $\prod\limits_{j=1}^{q} (\delta_j^\intercal x)^{\rho_j}$ be the product after collecting every class of collectable order-1 x-fragments so that after collecting, all δ_j's are distinct. Let $\Delta = [\delta_1, \cdots, \delta_q]$ be the column-binding of the individual δ_j's. Denote by $\delta_{(i)}$ the i-th row of Δ. Then any of $\delta_{(i)}$ not summing to at least 1 means *that* $\delta_{(i)} = 0$ which means the exhaustiveness condition of a slice is violated and the product can never be factorized into a product of any slices, not to mention any group of compatible slices. $\square$

Definition 4.26 (Total y-thickness of a superposition). For a superposition π with a compatibility axis passing through a point y, we define its 'total y-thickness' as $\bar{\tau}(\pi, y) := \sum_{k=1}^{g} \tau^{e_k}(y)$ where $\{e_k\}_1^g$ is a set of y-superposable slices whose product equals π.

Lemma 4.27. *Inheriting notations from Definition 4.26: The co-thickness superposed overhead each ion is equal to the total thickness of the superposition, that is, $\Delta\tau(\pi, y) = \bar{\tau}(\pi, y)\, \mathbf{1}_{n\times 1}$.*

Example 4.28. *Consider the superposition*

$$\pi_1 = \frac{x_1^4 x_2^6 x_3^8 x_4^{10} x_5^{12}(x_1 + x_2)^5(x_3 + x_4 + x_5)^{15}}{(x_1 + x_2 + x_3)^9(x_4 + x_5)^{11}}$$

in an earlier example where we have declared it should not be a fragment of any x-slice, but a product of three x-slices

$$
\begin{aligned}
e_1 &= x_1^4 x_2^6 x_3^8 x_4^{10} x_5^{12} \\
e_2 &= (x_1 + x_2)^5(x_3 + x_4 + x_5)^{15} \\
e_3 &= \frac{1}{(x_1 + x_2 + x_3)^9(x_4 + x_5)^{11}}
\end{aligned}
$$

superposable at $y = (2, 3, 4, 5, 6)^{\intercal}$ (or any scaling of this). The co-thickness achieved at y by slice e_1 is $\tau^{(1)}(y) = 2$. Similarly for e_2 and e_3, $\tau^{(2)}(y) = 1$ and $\tau^{(3)}(y) = -1$. Thus $e_1 \overset{y}{\leftrightarrows} e_2 \overset{y}{\leftrightarrows} e_3$ and their product π_1 is a justified superposition (of them). The event pattern matrix of e_1 is a 5×5 identity matrix; of e_2 and e_3 they are both 5×2 bit matrices. Therefore event pattern of the whole superposition π_1 is a 5×9 matrix. The total thickness of π_1 at y is $\bar{\tau}(\pi, y) = \tau^{(1)}(y) + \tau^{(2)}(y) + \tau^{(3)}(y) = 2$. Thus,

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
2 \\ 2 \\ 2 \\ 2 \\ 2 \\ 1 \\ 1 \\ -1 \\ -1
\end{bmatrix}
=
\begin{bmatrix}
2 \\ 2 \\ 2 \\ 2 \\ 2
\end{bmatrix}.
$$

4.3. Slicing Algorithm and Fundamental Conjecture.

Definition 4.29 (Slicing Algorithm). Given a superposition π. An algorithm $\mathfrak{A}$ inputs the initial syntax of π. The algorithm $\mathfrak{A}$ proceeds to find a finite set of mutually compatible slices $\{e_1, \ldots, e_g\}$ so that π is their superposition. Such an algorithm $\mathfrak{A}$ is called a *slicing algorithm* for π. A slicing algorithm is also said to be *slicing* π.

Remark 4.30. A slicing algorithm does not ionize nor unionize any fragments. A slicing algorithm does not change the event pattern—it merely redistributes the event counts.

Example 4.31. *Let $a, b, c, d > 0$ satisfy $a + b + c + d = 1$, then $\pi(a, b, c, d) = \frac{a^2 b^3 c^4 d^5}{(a+b)^4(c+d)^6}$ attains maximum at $(a_m, b_m, c_m, d_m) = (\frac{1}{10}, \frac{3}{20}, \frac{1}{3}, \frac{5}{12})$. We explain the method as following. Write*

$$
\begin{aligned}
\pi_{\max} = a^{as} b^{bs} c^{cs} d^{ds} \cdot (a+b)^{(a+b)u}(c+d)^{(c+d)u} \\
\cdot (a+b)^{(a+b)v} c^{cv} d^{dv} \cdot (c+d)^{(c+d)w} a^{aw} b^{bw}
\end{aligned}
$$

and note that we have 6 powers but total 7 variables, therefore one of the auxiliary variables s, u, v, w, must be redundant. We can set one of them to zero. Let's set $s = 0$ and generate the equations to match the exponents.

$$
\begin{cases}
aw = 2 \\
bw = 3 \\
cv = 4 \\
dv = 5 \\
(a+b)(u+v) = -4 \\
(c+d)(u+w) = -6 \\
a + b + c + d = 1
\end{cases}
\rightarrow
\begin{cases}
a = 2/w,\ b = 3/w,\ c = 4/v,\ d = 5/v \\
(u+v) = -4w/5, \qquad (u+w) = -2v/3 \\
\rightarrow v - w = 2v/3 - 4w/5 \\
\rightarrow v/3 = w/5 \rightarrow v = 3t,\ w = 5t \\
\dfrac{5}{w} + \dfrac{9}{v} = 1 \rightarrow t = 4 \\
\rightarrow v = 12,\ w = 20 \\
\implies a = \dfrac{1}{10},\ b = \dfrac{3}{20},\ c = \dfrac{1}{3},\ d = \dfrac{5}{12} \\
(\rightarrow u = -28)
\end{cases}
$$

Conjecture 4.32 (The Fundamental Conjecture of Superposition). *[FC] We give the following two equivalent statements of this conjecture:*

    a. Every product of slices is a superposition.

    b. Every product of order-1 fragments is a superposition, with respect to the smallest possible vector of ions implied from all of the order-1 fragment.

Remark 4.33. The conjecture is essentially asserting the existence statement in Definition e of superposition. We need the second version because it is the version more flexible and may be useful in a recursive implementation. We defer the proof of this conjecture for the future, with the strong faith that it is correct, at least under a few reasonable regularity conditions.

For now, we give a reason why the conjecture is plausible. We define the trivial slicing algorithm $\overset{\circ}{\mathfrak{A}}(\pi)$ with the conception that it will transform the claim of the conjecture into a claim of the existence of a solution to a polynomial system. That solution set is our goal: the compatibility axis of the superposition.

Overall, **the trivial slicing algorithm (TSA)** uses ionic fragments to fill the vacancies left by every unionic order-1 fragment to form a complete slice. In addition, there will usually be a slice completely made of ionic fragments.

Let x be the smallest possible vector of ions implied from the product. In the pattern-collected form, the product is written as $\pi = \prod_{k=1}^{M} (\delta_k^\mathsf{T} x)^{\rho_k}$. We then separate the set of ionic fragments from the set of non-ionic order-1 x-fragments and write

$$\pi = \prod_{i=1}^{n} x_i^{a_i} \prod_{j=1}^{Q} \left(\delta_j^\mathsf{T} x\right)^{b_j}$$

where a_i can take value 0 to indicate absence of x_i.

Let p be the point of intersection of the compatibility axis with the simplex having the same number of vertices as the length of x. We define the 0-th slice to be made all of ionic fragments, with co-thickness $\tau^{e_0}(p) = \tau_0$:

$$e_0 = \prod_{i=1}^{n} x_i^{\tau_0 p_i} \tag{4.1}$$

We then define the 1st - Qth slices, with co-thickness $\tau^{e_j}(p) = \tau_j$, to be

$$e_j = \left(\delta_j^\mathsf{T} x\right)^{\tau_j\left(\delta_j^\mathsf{T} p\right)} \cdot \prod_{i \in \{k \in \{1,\ldots,n\}:\delta_{jk}=0\}} x_i^{\tau_j p_i}, \qquad \forall j \in \{1,\ldots,Q\} \tag{4.2}$$

These $Q + 1$ slices are automatically superposable because they are defined to share the axis along p.

We need to determine the n components of the intersection p and the $Q + 1$ coefficients $\tau_0, \tau_1, \ldots, \tau_Q$, which is to say that we can solve the following system of equations of polynomials for the $n + Q + 1$ unknowns.

$$\begin{cases} b_1 &= \tau_1 \left(\delta_1^\mathsf{T} p\right) \\ &\vdots \\ b_Q &= \tau_Q \left(\delta_Q^\mathsf{T} p\right) \\ a_1 &= p_1 \left(\tau_0 + \sum_{j \in \{r \in \{1,\ldots,Q\}:\delta_{r1}=0\}} \tau_j\right) \\ &\vdots \\ a_n &= p_n \left(\tau_0 + \sum_{j \in \{r \in \{1,\ldots,Q\}:\delta_{rn}=0\}} \tau_j\right) \\ 1 &= p_1 + \cdots + p_n \end{cases} \tag{4.3}$$

In the lower a_i-block equations, the complicated summation of τ_js can be simplified if we use $\delta_{(r)}^\mathsf{T}$ to denote the r-th row of the $n \times Q$ matrix $\Delta = [\delta_1, \ldots, \delta_Q]$, and use the $Q \times 1$ column vector $\tau = (\tau_1, \ldots, \tau_Q)^\mathsf{T}$:

$$\begin{cases} a_1 &= p_1 \left(\tau_0 + \left(1_{1 \times Q} - \delta_{(1)}^\mathsf{T}\right)\tau\right) \\ &\vdots \\ a_n &= p_n \left(\tau_0 + \left(1_{1 \times Q} - \delta_{(n)}^\mathsf{T}\right)\tau\right) \end{cases} \tag{4.4}$$

These are $n + Q + 1$ polynomial equations written on exactly $n + Q + 1$ indeterminates. It remains for the future to show that these equations has at least one solution given any parameter triple (a, b, Δ).

Remark 4.34. Observe that there is a dual relationship in Eqn (4.3) (with the simplification of Eqn (4.4)) between (p, a) and (τ, b): the coordinates of p when expressed on the basis formed by the components of a is completely determined by τ (and τ_0); symmetrically, the coordinates of τ when expressed on the basis formed by the components of the vector b is completely determined by p. We will exploit this duality later when we device an iterative algorithm for solving the system. The last sum-to-one equation is a solitude constraint that does not interfere with the other equations. Its main use is to degenerate the solution set of p from originally an axis of points to a single point as the intersection of that original axis and the sum-to-one affine plane. Under it, a one-to-one correspondence exists between the point of intersection and the axis. Now that we are dealing with the art of solving polynomial equations, we are entering the realm of algebraic geometry. The TSA-yielded polynomial system itself does *not* imply positivity of x; and if we allow at the moment that x varies unrestrainedly, into any orthant it would like, then we see that the compatibility axis need not even intersect with the sum-to-one affine plane—the axis could just be parallel to the plane. This observation describes the reality of practice: we are given the system of polynomials; we start without the positivity constraint on x; we insisted on not using the positivity constraint until we have found the solution set and then we use it only to constrain the solution set. A reminder on the logical flow: we have not yet definitively associated the compatibility axis with the locus of extremities, despite the lurking temptation, which we will embrace next when a regularity criterion is ready.

4.4. Simplicial Regularity.

Definition 4.35. Some definitions related to 'simplex'.

a. (Simplex). Denote by
$$\mathbb{T}_{n-1} := \{(x_1, \ldots, x_n) \in [0, 1]^n : \mathbf{1}_{1 \times n} x = 1\} \tag{4.5}$$
the $n - 1$ dimensional simplex as the 'first' diagonal hyper-plane of the n dimensional unit cube. Then it is clear that its interior is
$$\mathbb{T}_{n-1}^{\circ} = \{(x_1, \ldots, x_n) \in (0, 1)^n : \mathbf{1}_{1 \times n} x = 1\}. \tag{4.6}$$
It is also clear that its boundary is
$$\partial \mathbb{T}_{n-1} = \{(x_1, \ldots, x_n) \in \mathbb{T}_{n-1} : \exists i \in \{1, \ldots, n\}, x_i = 0\}. \tag{4.7}$$

b. (Uniformly regular product of order-1 fragments). Let $x \in \mathbb{T}_{n-1}$. Let π be a product of order-1 simplicial x-fragments whose union is exhaustive, that is, the product is some x-superposition by [FC]. We say π is a *uniformly regular* product of order-1 x-fragments iff π converges everywhere on $\mathbb{T}_{n-1}$; in particular, π converges everywhere on $\partial \mathbb{T}_{n-1}$.

c. (The zero simplicial slice). Let $x \in \mathbb{T}_{n-1}$. We make the convenient definition
$$\prod_{i=1}^{n} x_i^{\infty} := 0 \tag{4.8}$$
and call it the *zero simplicial x-slice*.

Remark 4.36. For Nr.2, the definition of uniform regularity: it is essentially a continuity requirement, i.e., we are requiring/assuming the product to be a continuous function. Recall that all continuous function on a compact domain is uniformly continuous; hence the inclusion of "uniformly" in the name.

Criterion 4.37 (Criterion for uniform simplicial regularity for a superposition)*. Let $x \in \mathbb{T}_{n-1}$. Let Π be a product of order-1 simplicial x-fragments whose union is exhaustive. Then Π is uniformly regular iff the following two conditions are jointly satisfied*

a. *for every negatively powered order-1 fragment ι^- of Π, the powers of all order-1 fragments in Π covered by ι^- (including ι^-'s own power) sum to a non-negative real number.*

b. *for every <u>non-exhaustive</u> <u>union</u> U^- of some of these ι^-s such that U^- does not factorize Π, the powers of all order-1 fragments **of** Π covered by U^- (including the powers of all those ι^-s as they are covered by U^-, but not including the power of U^- as it is not an order-1 fragment of Π) sum to a non-negative real number.*

Proof. (only if) We will show that if either condition fails, then Π diverges somewhere on the boundary. Suppose $\Pi = \Pi'\iota^{-}\prod_{l=1}^{m}\kappa_l$ for which $\iota^{-} = (\delta_0^{\mathsf{T}}x)^{c_0}$ with $c_0 < 0$ and $\left\{\kappa_l = (\delta_l^{\mathsf{T}}x)^{c_l}, \forall l \in \{1,\ldots,m\} : \delta_l \leqslant \delta_0\right\}$ is the set of all other order-1 x-fragments of Π that are covered by ι^{-}. Then $\forall x \in \{x : \delta_0^{\mathsf{T}}x = 0\} \subset \partial\mathbb{T}_{n-1}$ we have $\iota^{-}\prod_{l=1}^{m}\kappa_l = 0^{c_0 + \sum_{l=1}^{m}c_l}$ which converges only if $c_0 + \sum_{l=1}^{m}c_l \geqslant 0$.

Suppose $\Pi = \Pi'\prod_{r=1}^{s}\iota_r^{-}\prod_{l=1}^{m}\kappa_l$ for which $U^{-} = \bigcup_{r=1}^{s}\iota_r^{-} = \bigcup_{r=1}^{s}(\delta_r^{\mathsf{T}}x)^{b_r} = \left[\left(\sum_{r=1}^{s}\delta_r\right)^{\mathsf{T}}x\right]^{\sum_{r=1}^{s}b_r}$ with every $b_r < 0$ and

$$\left\{\kappa_l = (\delta_l^{\mathsf{T}}x)_l^{c}, \forall l \in \{1,\ldots,m\} : \delta_l \leqslant \sum_{r=1}^{s}\delta_r\right\}$$

is the set of all other order-1 x-fragments of Π that are covered by U^{-}. Then $\forall x \in \left\{x : \left(\sum_{r=1}^{s}\delta_r\right)^{\mathsf{T}}x = 0\right\} \subset \partial\mathbb{T}_{n-1}$ we have $\prod_{r=1}^{s}\iota_r^{-}\prod_{l=1}^{m}\kappa_l = 0^{\sum_{r=1}^{s}b_r + \sum_{l=1}^{m}c_l}$ which converges only if $\sum_{r=1}^{s}b_r + \sum_{l=1}^{m}c_l \geqslant 0$.

(if) We will show that Π diverges only if either condition fails. It is clear that the product diverges only if $\mathfrak{M}$ *or* $\mathfrak{N}$, where $\mathfrak{M}=$ "the denominator is 0 and the numerator is not 0" and $\mathfrak{N}=$ "the numerator is a lower order of zero than is the denominator."

$\mathfrak{M}$ is true only if $\mathfrak{M}^{*}=$ "some of Π's negatively powered order-1 fragments are zero and none of its positively powered order-1 fragments is zero" is true. $\mathfrak{M}^{*}$ is true only if the first condition is false: if any negatively powered ι^{-} of Π covers any positively powered order-1 fragments then $\iota^{-} = 0 \implies$ numerator contains 0 as factor $\implies$ contradiction.

$\mathfrak{N}$ is true iff $\Pi = \frac{A}{B} = \frac{u0^{\alpha}}{v0^{\beta}}$ with $0 < \alpha < \beta$ and $uv \neq 0$. Write $B = v0^{\beta} = v\prod_{r=1}^{R}\frac{1}{\iota_r^{-}} = v\prod_{r=1}^{R}(\delta_r^{\mathsf{T}}x)^{-b_r}$ and $\beta = \sum_{r=1}^{R}(-b_r)$. Write $A = u0^{\alpha} = u\prod_{t=1}^{T}\iota_t^{+} = u\prod_{t=1}^{T}(\delta_t^{\mathsf{T}}x)^{c_t}$ and $\alpha = \sum_{t=1}^{T}c_t$. Then $\mathfrak{N}$ is true only if the second condition is false: Let $U^{-} = \bigcup_{r=1}^{R}\iota_r^{-} = \left[\left(\sum_{r=1}^{R}\delta_r\right)^{\mathsf{T}}x\right]^{\sum_{r=1}^{R}b_r}$ then the set of positively powered order-1 fragment covered by U^{-} in $\Pi = \frac{A}{B}$ is $H = \left\{\eta^{+} = (\delta^{\mathsf{T}}x)^{c} : \left(\sum_{r=1}^{R}\delta_r\right)^{\mathsf{T}}x = 0 \Rightarrow \delta^{\mathsf{T}}x = 0\right\}$ which is a subset of the ι_t^{+}s above. Then the sum of powers of order-1 x-fragments covered by U^{-} in Π is less than $\alpha - \beta < 0$, which means that $\mathfrak{N}$ is true only if the second condition is false. $\qquad\square$

Remark 4.38. The two conditions can be merged in to one: **for every non-exhaustive union U^{-} of some of the negatively powered order-1 fragments of Π , the powers of all order-1 fragments of Π covered by U^{-} sum to a non-negative real number.**

The criterion is not lean—an exponentially expensive enumeration underlies it. An algorithm that closely implements it will probably be non-polynomial time. In our own prototyping implementation, running time in seconds is approximately $T = \frac{N2^{1.0011546N}}{2^{14.2124253}} \sim N2^{N}$ where N is the number of ι^{-}s in Π and ≈ 13 is the maximal N for the program to finish below 10s. Since validation is so expensive, we will have to run the optimization without first validating that it is optimizable. The silver lining is that all superpositions resulting from sampling the ionized, coarsened, and conditional spaces are uniformly regular.

Note 4.39. Potential algorithmic optimization can be done if one considers the obvious (directed acyclic) graph underlying every superposition: Each node corresponds to a δ_k and we add a weight to each node to represent the corresponding count b_k; Edges are arrows emitting from a δ_k pointing to every $\delta_m \geqslant \delta_k$ (element-wise) so that there is no δ_l such that $\delta_m \geqslant \delta_l \geqslant \delta_k$. Then it is clear that those ionic nodes do not have any arrows entering them while those largest δ_k's do not emit any arrows. Such a graph represents the fragment covering structure of the superposition and we may call it the *graph of covering*.

Example 4.40. *We give the following examples to explain the proof above.*

a. $x \in \mathbb{T}_4$: $\frac{x_1^{100}x_2^{100}x_3^{100}x_4^{100}x_5(x_1+x_2)(x_3+x_4)^{20}}{(x_1+x_2+x_3)^{200}(x_2+x_3+x_4)^{220}}$

- *There are 3 <u>non-exhaustive</u> U^{-}:*
  - $(x_1 + x_2 + x_3)^{-200}$
    - *Covers $(x_1 + x_2 + x_3)^{-200}$, $(x_1 + x_2)^{1}$, x_1^{100}, x_2^{100}, x_3^{100}*

- * ⟶ *total counts are* $-200 + 1 + 100 + 100 + 100 \geqslant 0$
  - ○ $(x_2 + x_3 + x_4)^{-220}$
    - * *Covers* $(x_2 + x_3 + x_4)^{-220}$, $(x_3 + x_4)^{20}$, x_2^{100}, x_3^{100}, x_4^{100}
    - * ⟶ *total counts are* $-220 + 20 + 100 + 100 + 100 \geqslant 0$
  - ○ $(x_1 + x_2 + x_3 + x_4)^{-420} = (x_1 + x_2 + x_3)^{-200} \cup$ $(x_2 + x_3 + x_4)^{-220}$
    - * *Covers* $(x_1 + x_2 + x_3)^{-200}$, $(x_2 + x_3 + x_4)^{-220}$, $(x_1 + x_2)^1$, $(x_3 + x_4)^{20}$, x_1^{100}, x_2^{100}, x_3^{100}, x_4^{100}
    - * ⟶ *total counts are* $-200 - 220 + 1 + 20 + 100 + 100 + 100 + 100 \geqslant 0$
- *Therefore it is uniformly regular.*

b. $x \in \mathbb{T}_4$: $\frac{x_1^{100} x_2^{100} x_3^{100} x_4^{100} x_5 (x_1 + x_2)(x_3 + x_4)^{20}}{(x_1 + x_2 + x_3)^{202}(x_2 + x_3 + x_4)^{220}}$

- *There are 3 non-exhaustive U^-s like before.*
- *After repeating the analysis, we will find that for the last union U^-, $(x_1 + x_2 + x_3 + x_4)^{-422} = (x_1 + x_2 + x_3)^{-202} \cup (x_2 + x_3 + x_4)^{-220}$, all order-1 fragments covered by it have a negative total counts: $-202 - 220 + 1 + 20 + 100 + 100 + 100 + 100 < 0$.*
- *Therefore it is not uniformly regular. It diverges at the vertex of $x_5 = 1$.*
- *The graph of covering of this superposition is shown in Figure 4.2.*
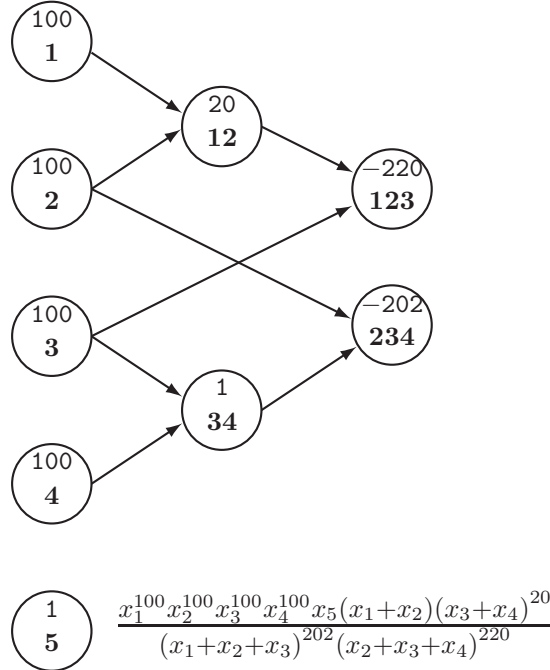

Figure 4.2: Graph of Covering for $\frac{x_1^{100} x_2^{100} x_3^{100} x_4^{100} x_5 (x_1 + x_2)(x_3 + x_4)^{20}}{(x_1 + x_2 + x_3)^{202}(x_2 + x_3 + x_4)^{220}}$

c. $x \in \mathbb{T}_3$: $\frac{x_1^{100} x_2^{100} x_3^{100} x_4^{100} (x_1 + x_2)(x_3 + x_4)^{20}}{(x_1 + x_2 + x_3)^{202}(x_2 + x_3 + x_4)^{220}}$

- *For this density we have only 2 non-exhaustive U^-s and both of them are well covered.*
  - ○ $(x_1 + x_2 + x_3)^{-202}$
  - ○ $(x_2 + x_3 + x_4)^{-220}$
- *Therefore this density is uniformly regular.*

Theorem 4.41. *Let $x \in \mathbb{T}_{n-1}$. Let $\pi = \prod\limits_{j=1}^{q} \left(\delta_j^{\mathsf{T}} x\right)^{\rho_j}$ be a simplicial x-superposition with only positive counts, i.e., $\forall j$, $\rho_j > 0$. Let p be the intersection of π's compatibility axis and $\mathbb{T}_{n-1}$. Asserting p's existence. Then p is a maximizer of $\pi(x)$.*

Proof. Let e be any of the mutually superposable slices of π. Then it is *sufficient* to prove that p is the unique maximizer of $e(x) = \prod_{j=1}^{q_1} \left(\gamma_j^\intercal x \right)^{d_j}$. This is an immediate corollary from Lemma 3.1:

$$
\begin{aligned}
e(x) &= \prod_{j=1}^{q_1} \left(\gamma_j^\intercal x \right)^{d_j} \\
&\leqslant H_1 \left[\sum_{j=1}^{q_1} \left(\gamma_j^\intercal x \right) \right]^{\sum_{j=1}^{q_1} d_j} = H_1 \left(\sum_{i=1}^{n} x_i \right)^{\sum_{j=1}^{q_1} d_j} = H_1 \\
&= \frac{\prod_{j=1}^{q_1} d_j^{d_j}}{\left(\sum_{j=1}^{q_1} d_j \right)^{\sum_{j=1}^{q_1} d_j}}
\end{aligned}
$$

whence equality is attained if $x = p$ (by definition, at p, the ratios of the exponents to bases are equalized). Note that "only if $x = p$" is not guaranteed, even though the maximal value attained at any such p must be the same constant given above. A counter example for "only if $x = p$" is $x_1(x_2 + x_3)$ for which a ridge is present. It is an uncomfortable fact that such a p may not be unique although $\pi(p)$ *is* unique. $\qquad\blacksquare$

Theorem 4.42 (p is a stationary point of π). *Let $x \in \mathbb{T}_{n-1}$. Let π be a simplicial x-superposition. Let p be the intersection of π's compatibility axis and $\mathbb{T}_{n-1}$. Then p is a stationary point of π.*

Proof. We start with separately processing π's numerator and denominator. First, for each order-1 fragment on the numerator, we complete it with a product of ionic fragments so that together their product becomes an x-slice maximized at p. At the same time we multiply this same product of ionic fragments to the denominator so that the whole fraction remains unchanged. Then the numerator has become a product of superposable x-slices all maximized at p. For each slice, we apply Lemma 3.1 in the <u>increasing</u> direction: to merge the bases from a product to a sum. For a slice, that sum is 1 which means we an replace it with a constant. Thus the whole numerator has become a product of constants. Moreover this product of constants maximizes the numerator.

On the denominator, for each order-1 fragment, we apply the <u>decreasing</u> direction of Lemma 3.1: to split an order-1 fragment into a product of ionic fragments, according to the proportion of p, times a constant. Thus the whole denominator will now become a product of all ionic fragments. Moreover this product of ionic fragments is smaller than the original denominator and it is minimized at p.

The fraction has now become a constant divided by a multinomial kernel multiplied by another constant: If we denote the fraction in this form by f, then

$$
\begin{cases}
\pi(x) \leqslant f(x) \\
\pi(p) = f(p)
\end{cases}
$$

because in both processing of the numerator and the denominator, we always increase the whole fraction while maintaining equality at p in every step. Moreover, f is stationarily minimal at p because its denominator, a multinomial kernel, is stationarily maximal at p. Thus p must be a stationary point of π, which is dominated by f and touches f only at p. $\qquad\blacksquare$

Remark 4.43. Theorem 4.42 should put the algebraic method developed here on equal footing to the calculus method of solving the likelihood equations.

Proposition 4.44 (Main inequality (conjectured)). *Let $x \in \mathbb{T}_{n-1}$. Let π be a uniformly regular simplicial x-superposition. Assume [FC] to be true. Let p be the intersection of π's compatibility axis and $\mathbb{T}_{n-1}$. Then $\pi(x) \leqslant \pi(p)$, $\forall x$, attaining equality (i.e. maximum) iff $x = p$.*

Remark 4.45. If true, this proposition will grant many properties including uniqueness of maximum of any such likelihood surface. This inequality is key to understand how the algebraic nature of equality attainment condition plays the necessary and sufficient role of the optimality condition, given the Fundamental Conjecture (i.e., existence of p) is true. In practice, the main inequality is broken down to two bridging inequalities, namely one to apply our seminal Lemma 3.1 to shrink the denominator by splitting it and the other to apply our seminal Lemma 3.1 again to aggrandize the numerator by merging its parts. Examples follow.

Example 4.46. *We randomly generate a few examples and show how the inequality, with its equality attainment condition, solves the optimization problem.*

a. $\frac{x_1^{10}x_2^{14}x_3^{17}}{(x_1+x_2)^4} \leqslant \frac{x_1^{10}x_2^{14}x_3^{17+ax_3}}{c_1 x_1^{ax_1}x_2^{ax_2}x_3^{ax_3}} = \frac{1}{c_1}x_1^{kx_1}x_2^{kx_2}x_3^{kx_3}, k > 0$

  - $x_1 : x_2 : x_3 = 10 : 14 : 20.4$

b. *This example unravels the optimization process of Π_1, for $x \in \mathbb{T}_4$.*

$$
\begin{aligned}
\Pi_1 &= \frac{x_1^4 x_2^6 x_3^8 x_4^{10} x_5^{12}(x_1+x_2)^5(x_3+x_4+x_5)^{15}}{(x_1+x_2+x_3)^9(x_4+x_5)^{11}} \\
&\leqslant \frac{x_1^4 x_2^6 x_3^8 x_4^{10} x_5^{12}(x_1+x_2)^5(x_3+x_4+x_5)^{15}}{\frac{1}{c_1}(x_1+x_2)^5 x_3^4 x_4^5 x_5^6} \\
&= c_1 x_1^4 x_2^6 x_3^4 x_4^5 x_5^6 (x_3+x_4+x_5)^{15} \\
&= c_1 \left[x_1^2 x_2^3 x_3^4 x_4^5 x_5^6 \right] \left[x_1^2 x_2^3 (x_3+x_4+x_5)^{15} \right] \\
&\leqslant c_1 \left[c_2(x_1+x_2+x_3+x_4+x_5)^{20} \right] \left[c_3(x_1+x_2+x_3+x_4+x_5)^{20} \right] \\
&= c_1 c_2 c_3 (x_1+x_2+x_3+x_4+x_5)^{40} \\
&= \frac{5^5 4^4}{9^9}\frac{5^5 6^6}{11^{11}}\frac{2^2 3^3 4^4 5^5 6^6}{20^{20}}\frac{2^2 3^3 15^{15}}{20^{20}}
\end{aligned}
$$

All equalities can be jointly attained at $[2,3,4,5,6]/20$. We see that compatibility and uniform regularity together enable the adsorption of all the denominator order-1 fragments into the numerator while Lemma 3.1 assures ascent of the whole expression in both the ionization of the denominator and the unionization in the numerator.

c. *Now consider a variation $\tilde{\Pi}_1$ of Π_1, still constructed from the same axis span$([2,3,4,5,6])$, for $x \in \mathbb{T}_4$:*

$$
\begin{aligned}
\tilde{\Pi}_1 &= \frac{x_1^4 x_2^6 x_3^8 x_4^{10} x_5^6 (x_3+x_4+x_5)^{15}}{(x_3+x_4)^9} \\
&\leqslant \frac{x_1^4 x_2^6 x_3^8 x_4^{10} x_5^6 (x_3+x_4+x_5)^{15}}{\frac{1}{c_1}x_3^4 x_4^5} \\
&= c_1 x_1^4 x_2^6 x_3^4 x_4^5 x_5^6 (x_3+x_4+x_5)^{15} \\
&= c_1 \left[x_1^2 x_2^3 x_3^4 x_4^5 x_5^6 \right] \left[x_1^2 x_2^3 (x_3+x_4+x_5)^{15} \right] \\
&\leqslant c_1 c_2 c_3 (x_1+x_3+x_3+x_4+x_5)^{40} \\
&= \frac{4^4 5^5}{9^9}\frac{2^2 3^3 4^4 5^5 6^6}{20^{20}}\frac{2^2 3^3 15^{15}}{20^{20}}
\end{aligned}
$$

attaining all equalities at $[2,3,4,5,6]/20$. Note that the ratio of $\tilde{\Pi}_1$ and Π_1 at every point along the compatibility axis is always $\frac{11^{11}}{5^5 6^6} > 1$.

d. *In this example we give the geometrical representation. Consider the superposition*

$$
\frac{x_1^4 x_2^{10} x_3^9 x_4^{24}(x_1+x_2)^6(x_1+x_3)^8(x_2+x_3)^{10}(x_3+x_4)^{14}}{(x_1+x_2+x_3)^6(x_2+x_3+x_4)^9}
$$

attaining maximum at $(p_1, p_2, p_3, p_4) = (0.1, 0.2, 0.3, 0.4)$. We represent it as assembly of fragments in Figure 4.3. There are positive thickness fragments (solid line) and negative thickness fragments (dotted line). There are two directions of operations: horizontal (join and split) and vertical (stack and section). Horizontal operations will increase or decrease total product. Whenever we join two positive thickness fragments whose patterns complements with each other, we increase the total product. Whenever we split a negative thickness fragment into a set of smaller fragments with the same negative thickness, we increase the total product. In short, in order to increase total product, we join solid-line components, and split dotted-line components. Vertical operations do not change total product. We may do whatever times of stacking or slicing without changing the total product. In Figure 4.3, numbers inside shapes are labels while numbers outside shapes are thicknesses (given we already know the compatibility axis). The mission is to use the above-described 4 operations:

  1. *joining solid-line fragments*

  2. *splitting dotted-line fragments*

$$Maximizer : (0.1, 0.2, 0.3, 0.4)$$

$$\frac{x_1^4 x_2^{10} x_3^9 x_4^{24} (x_1+x_2)^6 (x_1+x_3)^8 (x_2+x_3)^{10} (x_3+x_4)^{14}}{(x_1+x_2+x_3)^5 (x_2+x_3+x_4)^9}$$

Figure 4.3: Fragments, Slices, and Superposition: The Physical Picture

3. *stacking fragments*

4. *sectioning fragments*

to achieve complete slice with the positive thickness 70.

We can solve the puzzle as the following:

1. *Join fragment "12(thick 20)" and fragment "34(thick 20)" to form a complete slice "1234(thick 20)".*

2. *Split fragment "123(thick -10)" into "13(thick -10)" and "2(thick -10)"*

3. *Stack "13(thick -10)" on "13(thick 20)" to yield a "13(thick 10)"*

4. *Stack "2(thick -10)" on "2(thick 50)" to yield "2(thick 40)"*

5. *Split "234(thick -10)" into "23(thick -10)" and "4(thick -10)"*

6. *Stack "23(thick -10)" on "23(thick 20)" to yield "23(thick 10)"*

7. *Stack "4(thick -10)" on "4(thick 60)" to yield "4(thick 50)"*

8. *Now stack and join all fragments: "2 (thick 50)", "13(thick 10)", "1(thick 40)", "3(thick 30)", "23(thick 10)", "4(thick 50)", and "1234(thick 20)" to form "(1234thick 70)".*

Note 4.47. The extreme case would be to strip off 30 units of thickness from each of the 4 atomic fragments, vanishing x_3.

Corollary 4.48. *If [FC] is true, then two uniformly regular simplicial x-slices are superposable only if they share maximizers.*

Proposition 4.49 (The commutative semi-ring of uniformly regular simplicial slices). *Let e be a uniformly regular simplicial x-slice. Let W be the set of uniformly regular simplicial x-slices superposable with e. If both [FC] and Proposition 4.44 are true, then with the usual addition "+" and multiplication "·" between two functions, $(W, +, \cdot)$ is a commutative semi-ring, with $\prod_{i=1}^{n} x_i^{\infty}$ as the additive identity and $\prod_{i=1}^{n} x_i^{0}$ as the multiplicative identity.*

Proof. As we are using the usual addition and multiplication between two functions, associativity for the two operations and distributive law of multiplication on addition are readily hold. It is also obvious that the additive identity is $\prod_{i=1}^{n} x_i^{\infty} = 0$ and the multiplicative identity is $\prod_{i=1}^{n} x_i^{0} = 1$. Closures: Let f be a simplicial x-slice such that $f(x) \leqslant f(x^*)$. Let g be another simplicial x-slice such that $g(x) \leqslant g(x^*)$. Note that both f and g are positive functions. Then it is clear $(f + g)(x) = f(x) + g(x) \leqslant f(x^*) + g(x^*) = (f + g)(x^*)$ and $0 \leq (f \cdot g)(x) = f(x) \cdot g(x) \leqslant f(x^*) \cdot g(x^*) = (f \cdot g)(x^*)$. □

5 Algorithms

One quick application of the algebraic understanding of the relationship between ρ and p will lead us to devise a successful iteration. The intuition is that any point on the simplex should reconstruct its own ionic counts, if the unionic counts are fixed. Then by seeing how far these reconstructed ionic counts deviate from the actual ionic counts and how these deviations can vary, we get a piece of useful information for deciding how to proceed to the true maximum. A more specific description follows.

We first run the trivial slicing algorithm on π. By fixing the vector b and the bit matrix Δ, we can evaluate all except one of the auxiliary coefficients for the composite slices by

$$\tau_j(x) = \frac{b_j}{\delta_j^{\mathsf{T}} x}, \quad j = 1, 2, \ldots, q$$

or in vectorial form

$$\tau(x) = \frac{b}{\Delta^{\mathsf{T}} x}$$

where τ is the column vector $(\tau_1, \ldots, \tau_n)^{\mathsf{T}}$ and the division is understood as element-wise. It turns out that τ represents the vector of slices' thicknesses. The exception is the coefficient τ_0 for the ionic slice, which we will keep as an unknown variable to solve. We then reconstruct the ionic counts by

$$R_i(x, \tau_0) = x_i \left(\tau_0 + \sum_{j=1}^{q} (1 - \Delta_{ij}) \tau_j(x) \right)$$

or in vectorial form

$$R(x, \tau_0) = x * (\tau_0 \mathbf{1}_{n \times 1} + (\mathbf{1}_{n \times q} - \Delta) \tau(x))$$

where the vectorial multiplication $*$ is understood as element-wise, $\mathbf{1}_{n \times 1}$ the column vector of 1s, and $\mathbf{1}_{n \times q}$ the n by q matrix of 1s. We then find the individual reconstruction deviations by

$$d_i(x, \tau_0) = R_i(x, \tau_0) - a_i$$

or in vectorial form

$$d(x, \tau_0) = R(x, \tau_0) - a$$

and the sum of squared deviation

$$\varepsilon = d(x, \tau_0)^{\mathsf{T}} d(x, \tau_0).$$

It is clear that our objective is to make this ε zero, by choosing x and τ_0. Before delving into the details of the algorithm, we observe that τ_0 is in quadratic relation with ε, as quickly shown in the following.

Lemma 5.1 (Quadratic relationship between SSE and the zero-th auxiliary). $\varepsilon(\tau_0 | x) = a(x)\tau_0^2 + b(x)\tau_0 + c(x)$ *where*

$$a(x) = x^{\mathsf{T}} x,$$

$$b(x) = 2x^{\mathsf{T}} \{ diag(x)(\mathbf{1}_{n \times q} - \Delta)\tau(x) - a \},$$

$$\begin{aligned} c(x) = {} & \tau(x)^{\mathsf{T}} (\mathbf{1}_{q \times n} - \Delta^{\mathsf{T}}) diag(x)^{\mathsf{T}} diag(x)(\mathbf{1}_{n \times q} - \Delta)\tau(x) \\ & -2\tau(x)^{\mathsf{T}} (\mathbf{1}_{q \times n} - \Delta^{\mathsf{T}}) diag(x)^{\mathsf{T}} a + a^{\mathsf{T}} a \end{aligned}$$

Proof. It is clear that when x is fixed, $R(\tau_0|x)$ is linear in τ_0 with the form $\tau_0 x + \eta(x)$. Therefore $d(\tau_0|x) = \tau_0 x + \eta(x) - a$. Therefore $\varepsilon(\tau_0|x)$ is quadratic in τ_0 with second-order term $\tau_0^2 x^\mathsf{T} x$ and first-order term $2\tau_0 x^\mathsf{T}(\eta(x) - a)$, which expands to $2\tau_0 x^\mathsf{T}\{diag(x)(\mathbf{1}_{n \times q} - \Delta)\tau(x) - a\}$, and constant term $(\eta(x) - a)^\mathsf{T}(\eta(x) - a)$, which expands to the above-stated form of $c(x)$. □

Corollary 5.2 (A necessary condition for solution)**.** $\varepsilon = 0$ *only if* $b(x)^2 = 4a(x)c(x)$.

Algorithm 1: The Greedy Weaver Algorithm

Data: The ionic counts vector a, the unionic counts vector b, and the event pattern matrix Δ.
Ensure: a is a vector of positive reals; Δ is a 0-1 matrix, has the same number of rows as the length of a, and has the same number of columns as the length of b.
Result: The eigenestimate p.

1: $n \leftarrow \textbf{length}(a)$
2: $q \leftarrow \textbf{length}(b)$
3: $x \leftarrow a/\textbf{sum}(a)$
4: $\tau \leftarrow \textbf{function}(x)b \div (\Delta^\mathsf{T} x)$ // $\div$ is element-wise division
5: $\tau_0 \leftarrow \textbf{function}(x)\frac{x^\mathsf{T}\{a - \textbf{diag}(x)(\mathbf{1}_{n \times q} - \Delta)\tau(x)\}}{x^\mathsf{T} x}$ // according to Lemma 5.1
6: $d \leftarrow \textbf{function}(x)x * \{\tau_0(x)\mathbf{1}_{n \times 1} + (\mathbf{1}_{n \times q} - \Delta)\tau(x)\} - a$ // $*$ is elem-wise prod
7: $\varepsilon \leftarrow \textbf{function}(x)d(x)^\mathsf{T} d(x)$ // sum of squared error
8: $I \leftarrow \textbf{function}(x)\{$return the index of the component of $d(x)$ with the largest absolute value$\}$.
9: $\text{loopcount} \leftarrow 0$
10: **while** $\varepsilon(x) > 0.0000000000001$ **do**
11:    $i \leftarrow I(x)$ // the index of largest deviation
12:    /* perturb to learn the relationship between $x[i]$ and $d[i]$ using a interpolating parabola and trying to vanish $d[i]$ by setting $x[i]$ to a root of the parabola */
13:    /* making 3 x-values to interpolate a parabola: */
14:    $u \leftarrow \textbf{Array}\{0, x[i], 1.05x[i]\}$ // [] for array indexing, starts from 1
15:    $\text{temp} \leftarrow x$
16:    $\text{temp}[i] \leftarrow 1.05x[i]$
17:    // update the temp[n]
18:    $\text{temp}[n] \leftarrow 0$
19:    $\text{temp}[n] \leftarrow 1 - \textbf{sum}(\text{temp})$
20:    /* evaluate the corresponding y-values: */
21:    $v \leftarrow \textbf{Array}\{-a[i], d(x)[i], d(\text{temp})[i]\}$ // $d(x)[i]$ is ith component of $d(x)$
22:    /* find the coefs of the parabola $y = \alpha x^2 + \beta x + \gamma$ */
23:    $\gamma \leftarrow v[1]$ // array index starts from 1
24:    $\alpha \leftarrow \frac{u[3](v[2]-v[1])-u[2](v[3]-v[1])}{u[2]^2 u[3] - u[2]u[3]^2}$
25:    $\beta \leftarrow \frac{-u[3]^2(v[2]-v[1])+u[2]^2(v[3]-v[1])}{u[2]^2 u[3] - u[2]u[3]^2}$
26:    /* pick the suitable zero of the parabola, usu. the larger one: */
27:    $x[i] \leftarrow \frac{-\beta + \sqrt{\beta^2 - 4\alpha\gamma}}{2\alpha}$
28:    /* properly update $x[n]$ or normalize x, may add code */
29:    $x[n] \leftarrow 0$
30:    $x[n] \leftarrow 1 - \textbf{sum}(x)$
31:    $\text{loopcount} \leftarrow \text{loopcount} + 1$
32: **end**
33: $p \leftarrow x$ // The result

We now give our first greedy algorithm (Algorithm 1) named "Greedy Weaver". It is "greedy" because it carries out a sequence of conditional optimization on the "worst" x coordinate hoping that this sequence of conditional optimization will bring about the joint optimality. The pseudocode in Algorithm 1 is ready for a starting implementation.

Next we give a second algorithm. Recall in Remark 4.34 we have mentioned that there is a dual relationship between (a, x) and (b, τ). The foregoing algorithm has not utilized this observation. We specify the following Algorithm 2 named "Weaver" to exploit this observation. It removes the apparent "greedy" parts from "Greedy Weaver." The Weaver algorithm is the one used to solve the game in Example 2.4.

The "Greedy Weaver" and "Weaver" algorithms are children of the Reconstruction philosophy and the TSA method. Empirically if one puts the two in a `try...catch...` block in the order prioritizing Weaver over Greedy

---
Algorithm 2: The Weaver Algorithm

Data: The ionic counts vector a, the unionic counts vector b, and the event pattern matrix Δ.

Ensure: a is a vector of positive reals; Δ is a 0-1 matrix, has the same number of rows as the length of a, and has the same number of columns as the length of b.

Result: The eigenestimate p.

1: $n \leftarrow \textbf{length}(a)$
2: $q \leftarrow \textbf{length}(b)$
3: $x \leftarrow a/\textbf{sum}(a)$
4: $\tau \leftarrow \textbf{function}(x)b \div (\Delta^\mathsf{T} x)$ `// ÷ is element-wise division`
5: $\tau_0 \leftarrow \textbf{function}(x)\frac{x^\mathsf{T}\{a-\textbf{diag}(x)(1_{n\times q}-\Delta)\tau(x)\}}{x^\mathsf{T}x}$ `// according to Lemma 5.1`
6: $d \leftarrow \textbf{function}(x)x * \{\tau_0(x)1_{n\times 1} + (1_{n\times q} - \Delta)\tau(x)\} - a$ `// * is elem-wise prod`
7: $\varepsilon \leftarrow \textbf{function}(x)d(x)^\mathsf{T}d(x)$ `// sum of squared error`
8: $\text{loopcount} \leftarrow 0$
9: $\text{smallestSSE} \leftarrow \varepsilon(x)$
10: $\text{bestx} \leftarrow x$
11: $\text{SSE} \leftarrow \varepsilon(x)$
12: **while** $SSE > 0.0000000000001$ **do**
13:    $x \leftarrow a \div ((1_{n\times q} - \Delta)\tau(x) + \tau_0(x)1_{n\times 1})$ `// ÷ is elem-wise division`
14:    $x \leftarrow \frac{x}{\textbf{sum}(x)}$
15:    `// some optional bookkeeping:`
16:    **if** $SSE < smallestSSE$ **then**
17:      $\text{smallestSSE} \leftarrow \text{SSE}$
18:      $\text{bestx} \leftarrow x$
19:    **end**
20:    $\text{loopcount} \leftarrow \text{loopcount} + 1$
21:    $\text{SSE} \leftarrow \varepsilon(x)$
22: **end**
23: $p \leftarrow \text{bestx}$ `// The result`

Weaver and throw an `exception` carrying the `bestx` value whenever Weaver goes astray and use the `bestx` to initialize the Greedy Weaver, the pair work very well for a large proportion of inputs.

     try{
        Weaver with exception handling code
     } catch `exception`{
        Greedy Weaver with x initialized to `exception.bestx`
     }

We haven't mathematically studied their convergence properties or investigated what it means when the alliance fails on some input. But potentially the heavy reliance on the availability of ionic thickness (as TSA always uses ionic fragments to fill the vacancies) can cause divergence for some regular inputs.

     Two actual implementations in MATLAB and EXCEL VBA are available for download at the website accompanying this paper. The first is a minimal Excel VBA implementation specializes in portability and easy usage. The MATLAB implementation adds symbolic manipulations and requires version 2009a (7.8) or above to run.

6   Context

Some contextual works that are directly linked in topic with this paper are Hankin [2010], Ng et al. [2011], Hunter [2004]. Ng et al. [2011] is collects a number of Dirichlet related distributions; its Section 8.1 surveys results for the generalized Dirichlet Distribution (c.f. Dickey et al. [1987], Tian et al. [2003]), to which Theorem 4.41 of this paper applies.

6.1. The Hyper-Dirichlet Distribution of Hankin (2010). Counts are allocated into n categories, allowing unionic categories and the exclusion of certain categories. Each category possesses a characterizing attribute called "probability", collectively denoted by the simplicial vector **p**, to which the counts are proportional, up to an error expected to equal zero. The allocation process looks random by each individual count. But these counts collectively stabilizes to a statistical distribution whose conjugate distribution is motivated as the

"hyper-Dirichlet" distribution by Hankin [2010] where its density is stipulated as Hankin [2010, Eqn (5)]

$$f\left(\mathbf{p}\right) \propto \left(\prod_{i=1}^{k} p_i\right)^{-1} \prod_{G \in \wp(K)} \left(\sum_{i \in G} p_i\right)^{\mathcal{F}(G)}$$

where K is the set of positive integers not exceeding k, $\wp(K)$ is its power set, and $\mathcal{F}$ is a function that maps $\wp(K)$ to the real numbers. Hankin [2010] also prototypes a software package using a 1-D data structure based on the power set.[6] The Hyper-Dirichlet distribution (and its conjugate) is the model that pair with our technique. On the website accompanying this paper, we demonstrate the effectiveness of our Weaver algorithms pair on the massive volleyball data (c.f. Hankin [2010] Section 3.3 Team sports)[7] The greedy weaver handles this with only 87 iterations to achieve a precision of SSE=10^{-20}, significantly faster then all methods known to us that can correctly output a solution. For example, the Nelder-Mead Simplex method would take thousands of iterations to get close to this precision mainly because the region around MLE is too flattened (c.f. Hankin [2010] Section 3.3 for R code based on the `hyperdirichlet` package).

6.2. The Nelder-Mead Simplex Algorithm and The MM Algorithm of Hunter (2004). The Nelder-Mead Simplex optimization algorithm is used in software supplement to Hankin [2010] to find the mode of the hyper-Dirichlet distribution. The Nelder-Mead Simplex method (c.f. Nelder and Mead [1965], Lagarias et al. [1998]) senses the target function by inspecting its value at the finite collection of the vertices of a simplex. It adapts itself, per iteration, by reflection, shrinking, and expansion, to ensure that the lowest vertex is climbing. Usually brute-force search methods, though effective, are not efficient. But the Simplex method turns out to be very efficient and it is the default optimization method employed by the R function `stats :: optim()` in the basic package `stats` for general purpose optimization.

Hunter [2004] develops an MM (Minorization-Maximization) iteration for optimizing the Bradley-Terry model (c.f. Bradley and Terry [1952], David [1988]). We can understand the iteration as a Cauchy-sequence. Write the log-density with a Lagrange multiplier term $-\lambda\left(x_1 + \cdots + x_n - 1\right) \equiv 0$:

$$\ln f\left(x|a, b, \Delta\right) = -\ln c + \sum_{i=1}^{n} a_i \ln x_i + \sum_{j=1}^{q} b_j \ln\left(\delta_j^{\mathsf{T}} x\right) - \lambda\left(x_1 + \cdots + x_n - 1\right) \tag{6.1}$$

Write the likelihood equations with the Lagrange multiplier:

$$\begin{cases} 0 = \dfrac{a_i}{x_i} + \displaystyle\sum_{j=1}^{q} \dfrac{b_j \delta_{ij}}{\delta_j^{\mathsf{T}} x} - \lambda, \forall i = 1, \ldots, n \\ 1 = x_1 + \cdots + x_n \end{cases}$$

simplifying to

$$\begin{cases} x = \dfrac{a}{\lambda \mathbf{1}_{n \times 1} - \Delta\left(\frac{b}{\Delta^{\mathsf{T}} x}\right)} \\ 1 = \mathbf{1}_{n \times 1}^{\mathsf{T}} x \end{cases} \tag{6.2}$$

where any division between two vectors is element-wise. This gives rise to the following iteration of Hunter (2004):

$$x^{(t+1)} \leftarrow \frac{a}{\lambda \mathbf{1}_{n \times 1} - \Delta\left(\frac{b}{\Delta^{\mathsf{T}} x^{(t)}}\right)} \qquad \text{subject to } 1 = \mathbf{1}_{n \times 1}^{\mathsf{T}} x^{(t+1)} \tag{6.3}$$

Another facade of the same iteration is the transfer of optimization (c.f. [Lange et al., 2000]) of sum of a bunch of curved logarithms to sum of a bunch of linear slopes. In this particular case, the slopes are partial derivatives. But the partiality would not affect the overall optimality if one selectively linearize only those *negative* powers due to [Hunter, 2004, Eqn (9)]. Comparing to the Simplex Method, the MM iteration of Hunter (2004) would trade Simplex Method's vast admissible range of inputs for simplicity in implementation and improved speed.

6.3. Hessian matrix around the mode of a Hyper-Dirichlet density surface. The vintage Newton-Raphson method can also be used but as is well known the matrix inversion required by it may halt the iterations prematurely. We omit the details of the algorithmic mathematics here except that we have to mention the bonus of asymptotic variance estimate automatically embedded in Newton-Raphson. Separately, we derive the explicit form of the variance around the maximum as following.

---

[6]The problem with this data structure is that the data is usually sparse, leaving the powerset being most often unnecessarily exponential in time, space, and i/o.

[7]The full dataset was originally encapsulated in the R package `hyperdirichlet`'s `volleyball` data which contains an `hyperdirichlet` object called `vb_synthetic`; it has a 500×9 weaving grid.

Lemma 6.1. *Following [Ng et al., 2011, Eqn (8.18)]'s representation of [Hankin, 2010, Eqn (5)]:*

$$f\left(x|a,b,\Delta\right) \propto \prod_{i=1}^{n} x_i^{a_i} \prod_{j=1}^{q} \left(\delta_j^{\intercal} x\right)^{b_j} \tag{6.4}$$

the Hessian matrix of a hyper-Dirichlet log-density is given by

$$\mathcal{H}\left(x_{-n}\right) = -diag\left(\frac{a_1}{x_1^2}, \cdots, \frac{a_{n-1}}{x_{n-1}^2}\right) - \frac{a_n}{x_n^2} \mathbf{1}_{(n-1)\times(n-1)} - \sum_{j=1}^{q} \Psi_{n-1}\left(j\right) \tag{6.5}$$

where for $j = 1, \ldots, q$ and for $1 \leqslant i, k \leqslant n-1$:

$$\Psi_{n-1}\left(j\right) = \begin{bmatrix} \psi_{11}\left(j\right) & \cdots & \psi_{1,(n-1)}\left(j\right) \\ \vdots & \ddots & \vdots \\ \psi_{(n-1),1}\left(j\right) & \cdots & \psi_{(n-1),(n-1)}\left(j\right) \end{bmatrix}, \tag{6.6}$$

$$\psi_{ik}\left(j\right) = \frac{b_j\left(\delta_{ij} - \delta_{nj}\right)\left(\delta_{kj} - \delta_{nj}\right)}{\left(\delta_k^{\intercal} x\right)^2}. \tag{6.7}$$

Proof. Mainly derivations. The log-density is

$$\ln f\left(x|a,b,\Delta\right) = -c + \sum_{i=1}^{n} a_i \ln x_i + \sum_{j=1}^{q} b_j \ln\left(\delta_j^{\intercal} x\right)$$

The generic first partial derivative is

$$\frac{\partial \ln f\left(x\right)}{\partial x_i} = \frac{a_i}{x_i} - \frac{a_n}{x_n} + \sum_{j=1}^{q} \frac{b_j\left(\Delta_{ij} - \Delta_{nj}\right)}{\delta_j^{\intercal} x}$$

Then we can write the score vector as the following (this is a bonus).

$$\nabla_{n-1} \ln f\left(x|a,b,\Delta\right) = \underbrace{\begin{bmatrix} 1 & \cdots & 0 & -1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -1 \end{bmatrix}}_{(n-1)\times n} \left(\frac{a}{x} + \Delta\underbrace{\left(\frac{b}{\Delta^{\intercal} x}\right)}_{q\times 1}\right) \tag{6.8}$$

The generic second partial derivatives are

$$\frac{\partial^2 \ln f\left(x\right)}{\partial x_i^2} = -\frac{a_i}{x_i^2} - \frac{a_n}{x_n^2} - \sum_{j=1}^{q} \frac{b_j(\Delta_{ij} - \Delta_{nj})^2}{\left(\delta_j^{\intercal} x\right)^2}$$

$$\frac{\partial^2 \ln f\left(x\right)}{\partial x_i x_k} = -\frac{a_n}{x_n^2} - \sum_{j=1}^{q} \frac{b_j\left(\Delta_{ij} - \Delta_{nj}\right)\left(\Delta_{kj} - \Delta_{nj}\right)}{\left(\delta_j^{\intercal} x\right)^2}$$

To simplify notation, denote

$$\psi_{ik} = \sum_{j=1}^{q} \frac{b_j\left(\Delta_{ij} - \Delta_{nj}\right)\left(\Delta_{kj} - \Delta_{nj}\right)}{\left(\delta_j^{\intercal} x\right)^2} = \psi_{ki} \tag{6.9}$$

as the ik-th element of the symmetric matrix

$$\Psi_{n-1} = \begin{bmatrix} \psi_{11} & \cdots & \psi_{1,(n-1)} \\ \vdots & \ddots & \vdots \\ \psi_{(n-1),1} & \cdots & \psi_{(n-1),(n-1)} \end{bmatrix} \tag{6.10}$$

Then we can write the Hessian as

$$\mathcal{H}\left(x_{-n}\right) = -diag\left(\frac{a_1}{x_1^2}, \cdots, \frac{a_{n-1}}{x_{n-1}^2}\right) - \frac{a_n}{x_n^2} \mathbf{1}_{(n-1)\times(n-1)} - \Psi_{n-1} \tag{6.11}$$

$\square$

6.4. Algebraically solve the TSA polynomial system. Going back to the philosophical claim of this paper, i.e., the eigenreconstruction of ρ from p, we note that underpinning this philosophy is the system of polynomial equations that abide by the algebraic rules 2.1. The references for employing computational commutative algebra to attack the system include but not limited to: Pistone et al. [2000], Cox et al. [2006], Sturmfels [2002].

7 Software supplements

Online and Desktop implementations in Microsoft Excel 2007+ and MATLAB 2009a(version 7.8)+, are available at http://hku.hk/jdong/eigenstruct2013a.html.

References

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 00063444. URL http://www.jstor.org/stable/2334029.

David Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Undergraduate Texts in Mathematics. Springer, 2 edition, 2 2006.

H. A. David. *The Method of Paired Comparisons*. Oxford University Press, 2 edition, 1988.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 00359246. URL http://www.jstor.org/stable/2984875.

James M. Dickey, Jhy-Ming Jiang, and Joseph B. Kadane. Bayesian methods for censored categorical data. *Journal of the American Statistical Association*, 82(399):773–781, 1987. ISSN 01621459. URL http://www.jstor.org/stable/2288786.

L. R. Jr. Ford. Solution of a ranking problem from binary comparisons. *The American Mathematical Monthly*, 64(8):pp. 28–33, 1957. ISSN 00029890. URL http://www.jstor.org/stable/2308513.

Robin K. S. Hankin. A generalization of the Dirichlet distribution. *Journal of Statistical Software*, 33(11):1–18, 2 2010. ISSN 1548-7660. URL http://www.jstatsoft.org/v33/i11.

G. H. Hardy, J. E. Littlewood, and G. Pólya. *Inequalities*. Cambridge Mathematical Library. Cambridge University Press, 2 edition, 1988.

David R. Hunter. MM algorithms for generalized Bradley-Terry models. *Annals of Statistics*, 32(1):384–406, 2004. doi: 10.1214/aos/1079120141. URL http://projecteuclid.org/euclid.aos/1079120141.

J.C. Lagarias, J.A. Reeds, M.H. Wright, and P.E. Wright. Convergence properties of the Nelder–Mead simplex method in low dimensions. *SIAM Journal on Optimization*, 9(1):112–147, 1998.

Kenneth Lange, David R. Hunter, and Ilsoon Yang. Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, 9(1):1–20, 2000. doi: 10.1080/10618600.2000.10474858. URL http://amstat.tandfonline.com/doi/abs/10.1080/10618600.2000.10474858.

J. A. Nelder and R. Mead. A simplex method for function minimization. *The Computer Journal*, 7(4):308–313, 1 1965. doi: 10.1093/comjnl/7.4.308. URL http://comjnl.oxfordjournals.org/content/7/4/308.full.pdf+html.

Kai Wang Ng, Guo-Liang Tian, and Man-Lai Tang. *Dirichlet and Related Distributions: Theory, Methods and Applications*. Wiley Series in Probability and Statistics. Wiley, 5 2011.

Giovanni Pistone, Eva Riccomagno, and Henry P. Wynn. *Algebraic Statistics: Computational Commutative Algebra in Statistics*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CHAPMAN & HALL/CRC, 12 2000.

J. Michael Steele. *The Cauchy-Schwarz Master Class: An Introduction to the Art of Mathematical Inequalities*. Cambridge University Press, 4 2004.

Bernd Sturmfels. Solving systems of polynomial equations. In *GBMS Regional Conference Series in Mathematics*, volume 97, Providence, 2002. American Mathematical Society.

Guo-Liang Tian, Kai Wang Ng, and Zhi Geng. Bayesian computation for contingency tables with incomplete cell-counts. *Statistica Sinica*, 13(1):189–206, 2003.